



TUGAS AKHIR - KI141502

RANCANG BANGUN APLIKASI TERAPI PASCA STROKE UNTUK LATIHAN GERAK FUNGSIONAL TANGAN DENGAN MENGGUNAKAN LEAP MOTION CONTROLLER

SHAFLY NAUFAL ADIANTO
NRP 05111440000114

Dosen Pembimbing I
Wijayanti Nurul Khotimah, S.Kom., M.Sc.

Dosen Pembimbing II
Anny Yuniarti, S.Kom., M.Comp.Sc.

Departemen Informatika
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

(Halaman ini sengaja dikosongkan)



TUGAS AKHIR - KI141502

**RANCANG BANGUN APLIKASI TERAPI PASCA
STROKE UNTUK LATIHAN GERAK
FUNGSIONAL TANGAN DENGAN
MENGUNAKAN LEAP MOTION CONTROLLER**

SHAFLY NAUFAL ADIANTO
NRP 05111440000114

Dosen Pembimbing I
Wijayanti Nurul Khotimah, S.Kom., M.Sc.

Dosen Pembimbing II
Anny Yuniarti, S.Kom., M.Comp.Sc.

Departemen Informatika
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

(Halaman ini sengaja dikosongkan)



UNDERGRADUATE THESES - KI141502

**POST STROKE THERAPY APPLICATION
DESIGN FOR HAND MOTION FUNCTIONAL
REHEARSAL USING LEAP MOTION
CONTROLLER**

**SHAFLY NAUFAL ADIANTO
NRP 05111440000114**

**Supervisor I
Wijayanti Nurul Khotimah, S.Kom., M.Sc.**

**Supervisor II
Anny Yuniarti, S.Kom., M.Comp.Sc.**

**Department of Informatics
Faculty of Information Technology and Communication
Sepuluh Nopember Institute of Technology
Surabaya 2018**

(Halaman ini sengaja dikosongkan)

LEMBAR PENGESAHAN

RANCANG BANGUN APLIKASI TERAPI PASCA STROKE UNTUK LATIHAN GERAK FUNGSIONAL TANGAN DENGAN MENGGUNAKAN LEAP MOTION CONTROLLER

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Bidang Studi Interaksi Grafika dan Seni
Program Studi S-1 Departemen Informatika
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:

SHAFLY NAUFAL ADIANTO
NRP: 05111440000114

Disetujui oleh Pembimbing Tugas Akhir

1. Wijayanti Nurul Khotimah, S.Kom., M.Sc.
(NIP. 198603122012122004) (Pembimbing 1)
2. Anny Yuniarti, S.Kom., M.Comp.Sc.
(NIP. 198106222005012002) (Pembimbing 2)

SURABAYA
MEI 2018

(Halaman ini sengaja dikosongkan)

RANCANG BANGUN APLIKASI TERAPI PASCA STROKE UNTUK LATIHAN GERAK FUNGSIONAL TANGAN DENGAN MENGGUNAKAN LEAP MOTION CONTROLLER

Nama Mahasiswa : Shafly Naufal Adianto
NRP : 05111440000114
Jurusan : Informatika FTIK-ITS
Dosen Pembimbing 1 : Wijayanti Nurul Khotimah, S.Kom.,
M.Sc.
Dosen Pembimbing 2 : Anny Yuniarti, S.Kom., M.Comp.Sc.

ABSTRAK

Teknologi pendeteksi gerak tangan saat ini berhasil dikembangkan dengan baik pada perangkat lunak bernama Leap Motion Controller. Alat ini berfungsi sebagai alternatif kontrol input dalam interaksi antara manusia dengan komputer tanpa menggunakan sentuhan.

Ide yang digunakan dalam tugas akhir ini akan dibangun sebuah aplikasi terapi pasca stroke dengan menggunakan Unity dan Leap Motion SDK. Dengan tugas akhir ini diharapkan pengguna dapat melakukan terapi secara berkesinambungan dengan tidak bosan karena terapi pascastroke merupakan masa penting untuk memulihkan otot motorik pasien sebagai pengguna aplikasi.

Aplikasi ini telah diuji oleh ahli fisioterapi dengan harapan kekurangan yang ada dalam aplikasi ini dapat dikembangkan lebih lanjut di masa yang akan datang.

Kata kunci: Leap Motion, Pascastroke, Unity.

(Halaman ini sengaja dikosongkan)

POST STROKE THERAPY APPLICATION DESIGN FOR HAND MOTION FUNCTIONAL REHEARSAL USING LEAP MOTION CONTROLLER

Name : Shafly Naufal Adianto
NRP : 05111440000114
Department : Informatics FTIK-ITS
First Supervisor : Wijayanti Nurul Khotimah, S.Kom.,
M.Sc.
Second Supervisor : Anny Yuniarti, S.Kom., M.Comp.Sc.

ABSTRACT

The current motion detection technology has been successfully developed in software called Leap Motion Controller. This tool serves as an alternative input control in the interaction between humans and computers without using a touch.

The idea used in this final project will be a post stroke therapy application using Unity and Leap Motion SDK. With this final task is expected users can do therapy on an ongoing basis with no boredom because pascastroke therapy is an important time to restore motor muscle patients as users of the application.

This application has been tested by a physiotherapist in the hope that the deficiencies in this application can be further developed in the future.

Keywords : Leap Motion, Post Stroke, Unity.

(Halaman ini sengaja dikosongkan)

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirabbil'alamin, segala puji bagi Allah SWT yang telah melimpahkan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul:

“RANCANG BANGUN APLIKASI TERAPI PASCA STROKE UNTUK LATIHAN GERAK FUNGSIONAL TANGAN DENGAN MENGGUNAKAN LEAP MOTION CONTROLLER”

Selain ini, pada kesempatan ini penulis menghanturkan terima kasih sebesar-besarnya kepada pihak-pihak yang tanpa mereka, penulis tidak akan dapat menyelesaikan buku ini:

1. Allah SWT serta junjungan Nabi Muhammad SAW, karena limpahan rahmat dan karunia-Nya penulis dapat menyelesaikan Tugas Akhir dan juga perkuliahan di Teknik Informatika ITS.
2. Keluarga saya, yang tiada hentinya memberikan dukungan doa, nasehat, dan selalu mengingatkan penulis agar tidak lupa menjaga kesehatan saat menyelesaikan Tugas Akhir ini.
3. Ibu Wijayanti Nurul Khotimah, S.Kom., M.Sc. dan Ibu Anny Yuniarti, S.Kom., M.Comp.Sc.. selaku pembimbing I dan II yang telah membimbing dan memberikan motivasi, nasihat dan bimbingan dalam menyelesaikan Tugas Akhir ini.
4. Bapak dan Ibu dosen Departemen Informatika ITS yang telah banyak memberikan ilmu dan bimbingan yang tak ternilai harganya bagi penulis.

5. Rekan-rekan IGS Sahabat yang sudah membantu dan memberi hiburan saat penulis berada di laboratorium IGS.
6. Sahabat penulis dari group “Santai bisa unsend” yang sudah memberikan banyak pengalaman dan canda tawa selama masa perkuliahan.
7. Guru spiritual saya Rifat aka “Cina Lampung”.
8. Teman-teman saya yang jauh disana.
9. Teman-teman TC 2014, kakak-kakak TC 2012 & 2013 dan adik-adik TC 2015 & 2016 yang tidak dapat disebutkan satu per satu yang selalu membantu, menghibur, menjadi tempat bertukar ilmu serta pembelajaran baru dan berjuang bersama-sama penulis.
10. Serta semua pihak yang telah turut membantu penulis dalam menyelesaikan Tugas Akhir ini.

Penulis menyadari bahwa laporan Tugas Akhir ini masih memiliki banyak kekurangan. Oleh karena itu dengan segala kerendahan hati penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan penulis kedepannya. Selain itu, penulis berharap laporan Tugas Akhir ini dapat berguna bagi pembaca secara umum.

Surabaya, Mei 2018

DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
ABSTRAK.....	ix
ABSTRACT	xi
KATA PENGANTAR	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR	xix
DAFTAR TABEL.....	xxi
DAFTAR KODE SUMBER	xxiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Permasalahan	2
1.4 Tujuan	3
1.5 Manfaat.....	3
1.6 Metodologi	3
1.6.1 Penyusunan Proposal Tugas Akhir	3
1.6.2 Studi Literatur	4
1.6.3 Analisis dan Desain Perangkat Lunak	4
1.6.4 Perancangan dan Implementasi Perangkat Lunak	4
1.6.5 Pengujian dan Evaluasi.....	4
1.7 Sistematika Penulisan.....	5
BAB II DASAR TEORI.....	7
2.1 Leap Motion Controller.....	7
2.2 Unity (Game Engine)	8
2.3 Terapi Fungsional.....	8
2.4 Stroke	14
2.5 Laravel.....	14
2.6 Aplikasi Pembanding	15

BAB III ANALISIS DAN PERANCANGAN SISTEM17

3.1	Analisis Perangkat Lunak.....	17
3.1.1	Deskripsi Umum Perangkat Lunak.....	17
3.1.2	Spesifikasi Kebutuhan Fungsional Perangkat.....	18
3.2	Perancangan Perangkat Lunak	18
3.2.1	Model Kasus Penggunaan.....	19
3.2.2	Definisi Aktor	19
3.2.3	Rancangan Skenario Terapi	20
3.2.3.1	Tahap Dasar.....	21
3.2.3.2	Tahap Lanjut.....	23
3.2.4	Rancangan Antarmuka.....	26
3.2.4.1	Rancangan Antarmuka Menu Utama.....	26
3.2.4.2	Rancang Antarmuka Tahap Terapi.....	27
3.2.4.3	Model 3D.....	28
3.2.5	Arsitektur Umum Sistem	30
3.2.6	Rancang Proses Sistem	31
3.2.6.1	Rancang Proses Sistem Dalam Unity	31
3.2.6.2	Rancang Proses Sistem Dalam Laravel	33

BAB IV IMPLEMENTASI.....35

4.1	Lingkungan Implementasi Perangkat Lunak.....	35
4.2	Implementasi Skenario Terapi dan Antarmuka	36
4.2.1	Implementasi Non-terapi.....	36
4.2.1.1	Integrasi Leap Motion dengan Unity.....	36
4.2.1.2	Pilih User	37
4.2.1.3	Menu Utama	41
4.2.1.4	Pilih Level	42
4.2.1.5	Grafik.....	43
4.2.2	Implementasi Terapi	46
4.2.2.1	Menekan Tombol.....	46
4.2.2.2	Memegang Benda	50
4.2.2.3	Mengangkat Benda	51
4.2.2.4	Mengayunkan Benda	52
4.2.2.5	Menyalakan Lampu	53
4.2.2.6	Makan Dengan Sendok.....	55
4.2.2.7	Memindah Benda.....	56

4.2.2.8	Minum Dari Gelas	57
4.2.2.9	Memutar Kran	58
4.2.2.10	Membuka Pintu	60
BAB V UJI COBA DAN EVALUASI.....		63
5.1	Lingkungan Pengujian.....	63
5.2	Pengujian Aplikasi	64
5.2.1	Skenario Pengujian Fungsionalitas	64
5.2.2	Hasil Uji Coba Fungsionalitas	64
5.2.2.1	Uji Coba Membuat Akun	65
5.2.2.2	Uji Coba Latihan Terapi.....	65
5.2.2.3	Uji Coba Melihat Grafik.....	66
5.3	Konsultasi Dengan Ahli Fisioterapi	67
5.4	Evaluasi Pengujian	68
BAB VI KESIMPULAN DAN SARAN		71
6.1	Kesimpulan.....	71
6.2	Saran.....	71
DAFTAR PUSTAKA		73
LAMPIRAN.....		75
BIODATA PENULIS		77

(Halaman ini sengaja dikosongkan)

DAFTAR GAMBAR

Gambar 2.1 Leap Motion Controller	7
Gambar 2.2 Gerak anatomis pada memegang benda	9
Gambar 2.3 Gerak anatomis pada mencapai ketinggian tertentu	10
Gambar 2.4 Gerak anatomis pada makan dengan tangan	11
Gambar 2.5 Gerak anatomis pada menyisir rambut	12
Gambar 2.6 Gerak anatomis pada mencopot baju atau pakaian ..	13
Gambar 3.1 Menu Utama	26
Gambar 3.2 Tahap Terapi	27
Gambar 3.3 Rancangan Sederhana Arsitektur Aplikasi	31
Gambar 3.4 Flowchart Aplikasi	32
Gambar 3.5 Flowchart Laravel	33
Gambar 4.1 Model Data	36
Gambar 4.2 Hasil import <i>Package</i>	36
Gambar 4.3 Antarmuka Pilih User	37
Gambar 4.4 Antarmuka Menambah User Baru	40
Gambar 4.5 Antarmuka Menu Utama	41
Gambar 4.6 Antarmuka Pilih Level	42
Gambar 4.7 Antarmuka Pilih Level Tahap Dasar	42
Gambar 4.8 Antarmuka Pilih Level Tahap Lanjut	43
Gambar 4.9 Antarmuka Pilihan Grafik Tahap Dasar	44
Gambar 4.10 Antarmuka Pilihan Grafik Tahap Lanjut	44
Gambar 4.11 Antarmuka Grafik	45
Gambar 4.12 Antarmuka Menekan Tombol	46
Gambar 4.13 Antarmuka Memegang Benda	50
Gambar 4.14 Antarmuka Mengangkat Benda	51
Gambar 4.15 Antarmuka Mengayunkan Benda	52
Gambar 4.16 Antarmuka Menyalakan Lampu	53
Gambar 4.17 Antarmuka Makan Dengan Sendok	55
Gambar 4.18 Antarmuka Memindah Benda	56
Gambar 4.19 Antarmuka Minum Dari Gelas	58
Gambar 4.20 Antarmuka Memutar Kran	59
Gambar 4.21 Antarmuka Membuka Pintu	60

(Halaman ini sengaja dikosongkan)

DAFTAR TABEL

Tabel 2.1 Perbedaan dengan aplikasi sejenis	16
Tabel 3.1 Daftar Spesifikasi Kebutuhan Fungsional Perangkat Lunak.....	18
Tabel 3.2 Daftar Kasus Penggunaan	19
Tabel 3.3 Deskripsi Pengguna.....	19
Tabel 3.4 Gerakan Tahap Dasar	21
Tabel 3.5 Gerakan Tahap Lanjut	23
Tabel 3.6 Spesifikasi Atribut Antarmuka Menu Utama	26
Tabel 3.7 Spesifikasi Atribut Antarmuka Tahap Terapi	27
Tabel 3.8 Daftar Objek.....	28
Tabel 5.1 Lingkungan Pengujian Sistem.....	63
Tabel 5.2 Uji Coba Membuat Akun	65
Tabel 5.3 Uji Coba Latihan Terapi.....	66
Tabel 5.4 Uji Coba Melihat Grafik	66
Tabel 5.5 Hasil Evaluasi Skenario	68

(Halaman ini sengaja dikosongkan)

DAFTAR KODE SUMBER

Kode Sumber 4.1 <i>Function</i> get_user.....	37
Kode Sumber 4.2 Mengambil Data.....	38
Kode Sumber 4.3 Mengolah Data	38
Kode Sumber 4.4 <i>Generate</i> Button	39
Kode Sumber 4.5 Menyimpan Id User.....	40
Kode Sumber 4.6 <i>function</i> create	40
Kode Sumber 4.7 Load Scene	41
Kode Sumber 4.8 <i>function</i> grafik2	46
Kode Sumber 4.9 Collision jari dan tombol.....	47
Kode Sumber 4.10 Menyalakan <i>timer</i>	48
Kode Sumber 4.11 Menyimpan Hasil	49
Kode Sumber 4.12 <i>function</i> update	49
Kode Sumber 4.13 Mengecek Genggaman	51
Kode Sumber 4.14 Mengecek Lokasi	52
Kode Sumber 4.15 Men- <i>generate</i> karakter	54
Kode Sumber 4.16 Input Karakter	54
Kode Sumber 4.17 Pengecekan String	55
Kode Sumber 4.18 Cek <i>Collision</i> Sendok.....	56
Kode Sumber 4.19 Cek Posisi Buku	57
Kode Sumber 4.20 Cek <i>Collision</i> Gelas.....	58
Kode Sumber 4.21 Putar Kran	60
Kode Sumber 4.22 Rotasi Pintu	61

(Halaman ini sengaja dikosongkan)

BAB I

PENDAHULUAN

1.1 Latar Belakang

Stroke merupakan kerusakan otak yang terjadi akibat terpotongnya aliran darah dan oksigen ke otak. Dua juta sel otak mati setiap menit selama stroke yang jika tidak ditangani dengan cepat akan memperluas kerusakan otak sehingga bisa menyebabkan cacat tubuh semakin parah atau bahkan menyebabkan kematian [1]. Sel-sel otak yang sudah mati tidak dapat disembuhkan lagi karena jaringan otak tidak memiliki kapasitas untuk melakukan regenerasi [2]. Namun dalam beberapa kasus, penderita stroke yang ditangani dengan cepat dan melakukan rehabilitasi dengan tepat, memiliki tingkat pemulihan gerak yang hampir sempurna walaupun mayoritas penderita stroke mengalami cacat permanen [3].

Rehabilitasi yang dilakukan penderita stroke juga sebatas pada gerak fungsional, bukan gerak aktual. Gerak aktual adalah pergerakan otot yang dilakukan secara sederhana seperti menekuk siku, menekuk lutut dan mengepalkan tangan, sedangkan gerak fungsional adalah gerak aktual untuk melakukan aktivitas sesuai yang diharapkan sebagai contoh menyisir rambut, minum dari gelas dan memutar kran air [4], [5].

Penderita stroke bisa dikatakan pulih jika mereka mampu melakukan gerak fungsional secara mandiri [4]. Maksud dari mandiri disini adalah mereka bisa melakukan kegiatan sehari-hari tanpa memerlukan bantuan orang lain bukan melakukan kegiatan dengan gerak yang sempurna seperti orang pada umumnya. Untuk bisa pulih, penderita stroke harus menjalani rehabilitasi secara rutin dimana setiap pertemuan untuk mengunjungi fisioterapi memerlukan biaya yang tidak murah. Hal itu mungkin tidak terlalu seberapa jika

rehabilitasi hanya dilakukan dalam waktu singkat namun, kita tidak tahu kapan rehabilitasi dapat dicukupkan.

Pemanfaatan teknologi yang tepat bisa menjadi salah satu solusi atas permasalahan tersebut. Dalam tugas akhir ini, teknologi yang akan digunakan adalah Leap Motion Controller dimana alat tersebut mampu menerima input dan mengontrol sebuah aplikasi hanya dengan menggunakan sensor gerak tangan. Alat ini juga dapat digunakan untuk terapi gerak fungsional yang berhubungan dengan tangan pada penderita pasca stroke dimana penderita tidak perlu mengeluarkan biaya setiap kali melakukan terapi.

1.2 Rumusan Masalah

Tugas Akhir ini mengangkat beberapa rumusan masalah sebagai berikut:

1. Bagaimana menyusun skenario terapi yang sesuai dengan terapi dalam kehidupan nyata?
2. Bagaimana rancang bangun dan implementasi terapi gerak fungsional pada jari tangan pasca stroke?

1.3 Batasan Permasalahan

Permasalahan yang dibahas pada Tugas Akhir ini memiliki batasan sebagai berikut:

1. Aplikasi yang dibuat merupakan aplikasi *desktop*.
2. Lingkungan pengembangan yang digunakan menggunakan aplikasi Unity 3D dengan bahasa pemrograman C#.
3. Aplikasi dibuat menggunakan Leap Motion SDK.
4. Menggunakan organ gerak lengan tangan.
5. Digunakan pada pasien pascastroke yang tidak mengalami kelumpuhan tangan total dan dapat menggerakkan jari serta lengan bagian bawah.
6. Gerakan fungsional pada aplikasi ini hanya sebatas minum dari gelas, makan dengan sendok, memutar kran

air, membuka pintu, menyalakan lampu dan meletakkan barang di rak.

7. Gerak aktual pada aplikasi ini hanya sebatas menekan tombol, memegang benda, mengangkat benda dan mengayunkan benda.

1.4 Tujuan

Tujuan dari pembuatan Tugas Akhir ini yaitu membangun suatu perangkat lunak yang dapat menjadi salah satu media terapi gerak fungsional tangan bagi pasien pasca stroke secara virtual. Tujuan lain dari pembuatan Tugas Akhir ini yaitu melakukan pengembangan Leap Motion Controller khususnya sebagai alat bantu medis.

1.5 Manfaat

Manfaat yang diharapkan dari pengembangan aplikasi ini diantaranya :

1. Terciptanya sebuah aplikasi yang dapat menjalankan terapi pasca stroke secara virtual dengan memanfaatkan Leap Motion Controller.
2. Dapat menjadi referensi untuk penelitian lebih lanjut mengenai pengembangan Leap Motion Controller dalam bidang medis.

1.6 Metodologi

Pembuatan Tugas Akhir ini dilakukan dengan menggunakan metodologi sebagai berikut:

1.6.1 Penyusunan Proposal Tugas Akhir

Proposal tugas akhir ini berisi deskripsi pendahuluan tugas akhir yang akan dikerjakan. Bagian pendahuluan terdiri dari latar belakang tugas akhir, rumusan masalah, batasan masalah, tujuan pengerjaan tugas akhir, dan manfaat dari hasil pengerjaan tugas

akhir. Selain itu dalam proposal tugas akhir juga dijabarkan tinjauan pustaka yang digunakan sebagai referensi pendukung pengerjaan tugas akhir. Dalam proposal ini juga terdapat penjelasan mengenai metodologi yang dipakai, mulai dari tahap penyusunan proposal hingga penyusunan buku tugas akhir. Terdapat juga sub bab jadwal kegiatan yang menjelaskan jadwal pengerjaan tugas akhir.

1.6.2 Studi Literatur

Dalam studi literatur penulis akan mempelajari beberapa referensi terkait topik tugas akhir. Beberapa referensi tersebut adalah mengenai Leap Motion Controller, Leap Motion SDK, Unity, dan berkonsultasi dengan fisioterapi.

1.6.3 Analisis dan Desain Perangkat Lunak

Aplikasi ini memiliki fitur-fitur sebagai berikut :

1. Melakukan Terapi
2. Melihat grafik

1.6.4 Perancangan dan Implementasi Perangkat Lunak

Aplikasi ini akan dibangun menggunakan Game Engine Unity 3D Free, dengan bahasa pemrograman C#. Aplikasi ini juga memanfaatkan Leap Motion SDK yang dihubungkan dengan Leap Motion Controller.

1.6.5 Pengujian dan Evaluasi

Tahap pengujian dan evaluasi berisi pengujian aplikasi dan evaluasi berdasarkan hasil pengujian. Pengujian akan dilakukan oleh sampel pengguna, yaitu pasien pasca stroke yang memenuhi kriteria.

1.7 Sistematika Penulisan

Buku Tugas Akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan Tugas Akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku Tugas Akhir terdiri atas beberapa bagian seperti berikut ini.

Bab I Pendahuluan

Bab ini berisi latar belakang masalah, tujuan dan manfaat pembuatan Tugas Akhir, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan Tugas Akhir.

Bab II Dasar Teori

Bab ini membahas beberapa teori penunjang yang berhubungan dengan pokok pembahasan dan mendasari pembuatan Tugas Akhir ini.

Bab III Analisis dan Perancangan Sistem

Bab ini membahas mengenai perancangan perangkat lunak. Perancangan perangkat lunak meliputi perancangan data, arsitektur, proses dan perancangan antarmuka pada aplikasi.

Bab IV Implementasi

Bab ini berisi implementasi dari perancangan perangkat lunak.

Bab V Pengujian dan Evaluasi

Bab ini membahas pengujian dari aplikasi yang dibuat dengan melihat keluaran yang dihasilkan oleh aplikasi dan evaluasi untuk mengetahui kemampuan aplikasi serta mengetahui penilaian aspek kegunaan (*usability*) dari perangkat lunak.

Bab VI Kesimpulan dan Saran

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan. Bab ini membahas saran-saran untuk pengembangan sistem lebih lanjut.

Daftar Pustaka

Merupakan daftar referensi yang digunakan untuk mengembangkan Tugas Akhir.

Lampiran

Merupakan bab tambahan yang berisi daftar istilah yang penting pada aplikasi ini.

BAB II

DASAR TEORI

Bab ini berisi pembahasan mengenai teori-teori dasar yang digunakan dalam Tugas Akhir. Teori-teori tersebut diantaranya adalah Rancang Bangun Perangkat Lunak, *Leap Motion Controller*, Unity, Terapi Fungsional, Stroke, Laravel dan beberapa teori lain yang mendukung pembuatan Tugas Akhir.

2.1 Leap Motion Controller

Leap Motion Controller merupakan suatu perangkat yang dikembangkan oleh Leap Motion, Inc pada tahun 2008 oleh David Holz [6]. Leap Motion Controller sendiri digunakan sebagai input dari komputer tanpa sentuh. Bisa dikatakan bahwa Leap Motion ini merupakan pengganti mouse, karena mempunyai tujuan dan fungsi yang sama. Pada tahun 2010, untuk pertama kalinya Leap Motion Controller ini diperkenalkan pada publik . Leap Motion merupakan sebuah perangkat sensor hardware komputer yang mendukung gerakan tangan dan jari sebagai input, tetapi tidak memerlukan kontak tangan ataupun sentuhan. Bentuk leap motion dapat dilihat pada Gambar 2.1



Gambar 2.1 Leap Motion Controller

2.2 Unity (Game Engine)

Unity merupakan sebuah game engine yang dikembangkan oleh Unity Technologies. Unity dapat menciptakan game ke dalam beberapa sistem operasi sekaligus. Antara lain: Windows Phone, Android, iOS, Windows 8, OSX, Blackberry 10, Playstation 3, Playstation 4, XBOX, dan sebagainya. Game yang dapat dibuat dengan Unity ini bisa dalam bentuk 3D atau 2D, tergantung pada developer game tersebut. Unity mampu mengubah gambar statis menjadi animasi yang dapat dimainkan. Maka dari itu banyak partner yang menggunakan Unity, antara lain: Microsoft, Sony, Qualcomm, Blackberry, Samsung, dan Nintendo [7].

2.3 Terapi Fungsional

Terapi fungsional merupakan latihan penguatan dengan menggunakan beban dari dalam tubuh sendiri [8]. Namun agar dapat memaksimalkan kontraksi dari otot tersebut maka diperlukan beban eksternal. Dan terfokus pada latihan beberapa otot (lebih dari satu otot yang ikut bekerja) yang menggantikan kerja dari otot yang diisolasi pada suatu jenis latihan atau gerakan dan juga beberapa sendi. Selain itu pada latihan ini juga harus mengintegrasikan semua aspek dalam melakukan gerakan seperti berikut :

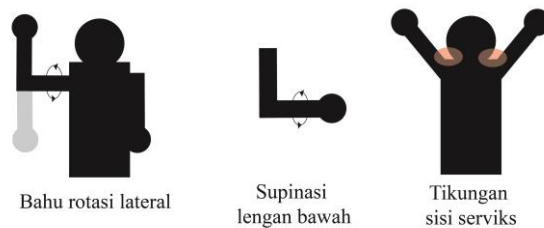
1. Balance
2. Performa Otot
3. Daya tahan kardiopulmonal
4. Mobilitas dan fleksibilitas
5. Kordinasi NM control
6. Stabilitas

Adapun latihan fungsional yang disarankan dan sesuai dengan aspek yang disebutkan diatas antara lainnya :

1. Memegang benda

Latihan ini menggunakan beberapa gerakan anatomis tubuh yaitu bahu rotasi lateral, supinasi lengan bawah, dan tikungan sisi serviks. Bahu rotasi lateral merupakan gerakan bahu

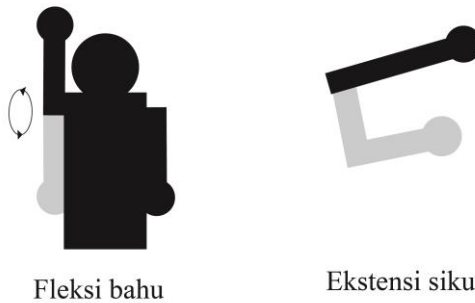
yang berputar menjauhi titik tengah tubuh. Supinasi lengan bawah merupakan gerakan rotasi lengan bawah ke arah luar tubuh. Tikungan sisi serviks merupakan gerakan yang terjadi antara leher dan bahu [9]. Gerakan-gerakan anatomis tubuh yang digunakan pada latihan memegang benda dapat dilihat pada Gambar 2.2.



Gambar 2.2 Gerak anatomis pada memegang benda

2. Mencapai ketinggian tertentu

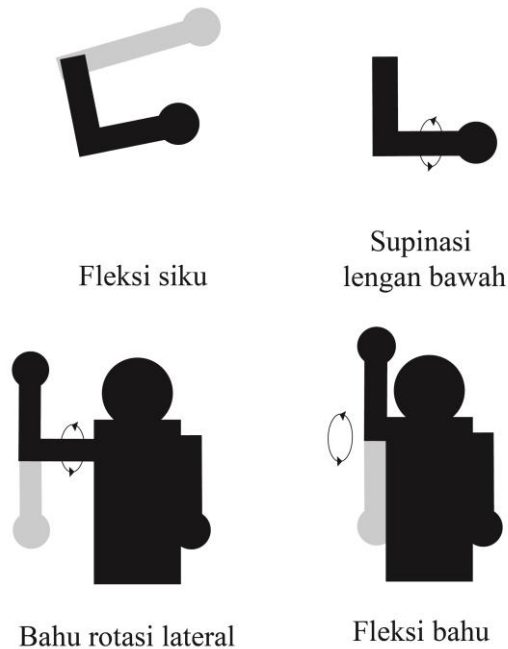
Latihan ini menggunakan 2 gerakan anatomis tubuh yaitu fleksi bahu dan ekstensi siku. Fleksi bahu merupakan gerak sendi yang memperkecil sudut sendi yang berada pada bahu. Ekstensi siku merupakan gerak meluruskan ketika sendi bergerak dari posisi fleksi kembali ke posisi normal [9]. Gerakan-gerakan anatomis tubuh yang digunakan pada latihan mencapai ketinggian dapat dilihat pada Gambar 2.3.



Gambar 2.3 Gerak anatomis pada mencapai ketinggian tertentu

3. Makan dengan tangan

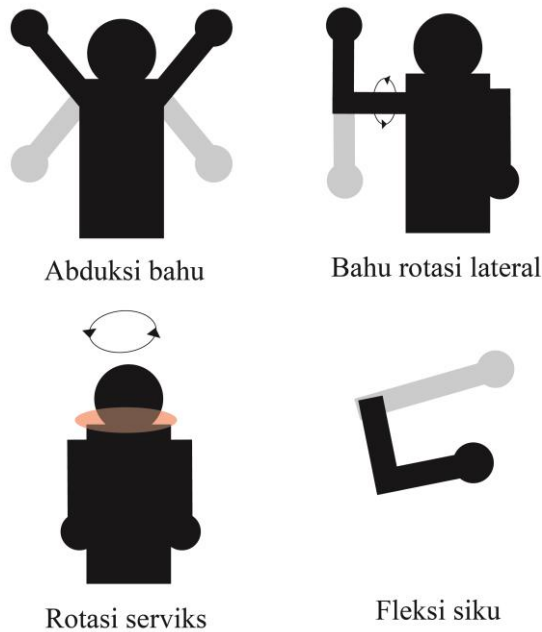
Latihan ini menggunakan beberapa gerakan anatomis tubuh yaitu fleksi siku, supinasi lengan bawah, fleksi bahu, dan bahu rotasi lateral. Fleksi siku merupakan gerak sendi yang memperkecil sudut sendi yang berada pada siku. Supinasi lengan bawah merupakan gerakan rotasi lengan bawah ke arah luar tubuh. Fleksi bahu merupakan gerak sendi yang memperkecil sudut sendi yang berada pada bahu. Bahu rotasi lateral merupakan gerakan bahu yang berputar menjauhi titik tengah tubuh [9]. Gerakan-gerakan anatomis tubuh yang digunakan pada latihan makan dengan tangan dapat dilihat pada Gambar 2.4.



Gambar 2.4 Gerak anatomis pada makan dengan tangan

4. Menyisir rambut

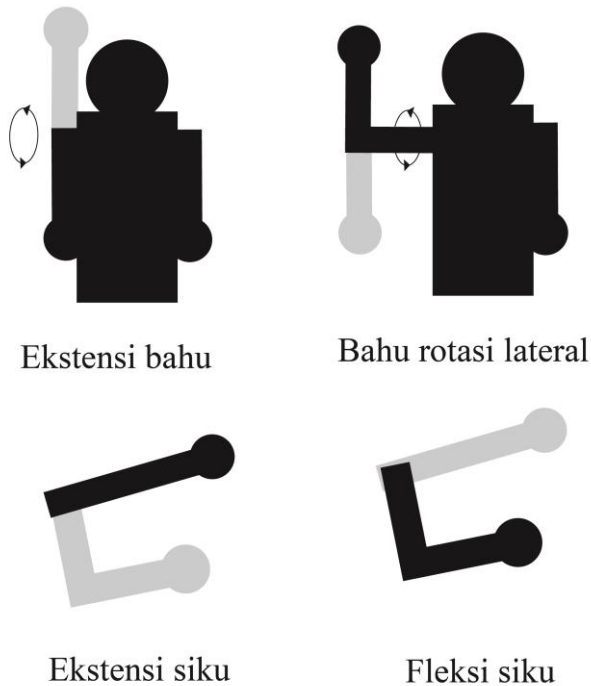
Latihan ini menggunakan beberapa gerakan anatomis tubuh yaitu abduksi bahu, bahu rotasi lateral, fleksi siku, dan rotasi serviks. Abduksi bahu merupakan gerakan ke samping, menjauhi tubuh. Bahu rotasi lateral merupakan gerakan bahu yang berputar menjauhi titik tengah tubuh. Fleksi siku merupakan gerak sendi yang memperkecil sudut sendi yang berada pada siku. Rotasi serviks merupakan perputaran sendi yang terdapat pada leher [9]. Gerakan-gerakan anatomis tubuh yang digunakan pada latihan menyisir rambut dapat dilihat pada Gambar 2.5.



Gambar 2.5 Gerak anatomis pada menyisir rambut

5. Mencopot baju atau jaket

Latihan ini menggunakan beberapa gerakan anatomis tubuh yaitu ekstensi bahu, bahu rotasi lateral, fleksi siku dan ekstensi siku. Ekstensi bahu merupakan gerak meluruskan sendiri atau kembali pada posisi normal. Bahu rotasi lateral merupakan gerakan bahu yang berputar menjauhi titik tengah tubuh. Fleksi siku merupakan gerak sendi yang memperkecil sudut sendi yang berada pada siku. Ekstensi siku merupakan gerak meluruskan ketika sendi bergerak dari posisi fleksi kembali ke posisi normal [9]. Gerakan-gerakan anatomis tubuh yang digunakan pada latihan mencopot baju atau pakaian dapat dilihat pada Gambar 2.6.



Gambar 2.6 Gerak anatomis pada mencopot baju atau pakaian

Dari saran latihan gerak fungsional yang sudah dipaparkan diatas, tidak semua jenis latihan dapat diterapkan dalam tugas akhir ini. Sebagai contoh gerakan menyisir rambut sulit untuk diterapkan karena Leap Motion tidak bisa mendeteksi tangan saat posisi tangan berada di belakang kepala. Mencopot baju dan jaket juga tidak bisa diterapkan karena cakupan Leap Motion tidak selebar saat manusia membuka baju atau jaket.

Terapi gerak fungsional yang dapat divirtualkan antara lain memegang benda, mencapai ketinggian dan makan dengan tangan. Gerakan itu akan menjadi dasar dari tugas akhir ini dan akan ditambahkan beberapa gerakan yang mengacu pada gerakan tersebut.

2.4 Stroke

Stroke merupakan kerusakan otak yang terjadi akibat terpotongnya aliran darah dan oksigen ke otak. Dua juta sel otak mati setiap menit selama stroke yang jika tidak ditangani dengan cepat akan memperluas kerusakan otak sehingga bisa menyebabkan cacat tubuh semakin parah atau bahkan menyebabkan kematian [1]. Sel-sel otak yang sudah mati tidak dapat disembuhkan lagi karena jaringan otak tidak memiliki kapasitas untuk melakukan regenerasi [2].

Karena otak tidak memiliki kapasitas untuk melakukan regenerasi, hal itu menyebabkan pasien tidak memiliki harapan untuk sembuh total (tergantung tingkat keparahan stroke yang diderita) [4]. Pasien hanya memiliki harapan untuk pulih dalam artian pasien bisa melakukan hal-hal dalam kehidupan sehari-hari secara mandiri. Pasien bisa pulih jika pasien memiliki semangat yang tinggi untuk hidup dan rutin untuk melakukan terapi gerak fungsional.

2.5 Laravel

Laravel adalah sebuah framework PHP yang dirilis dibawah lisensi MIT, dibangun dengan konsep MVC (model view controller) [9]. Laravel adalah pengembangan website berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu.

MVC adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi berdasarkan komponen- komponen aplikasi, seperti : manipulasi data, controller, dan user interface.

2.6 Aplikasi Pemandangan

Teknologi pendeteksi gerak tangan saat ini berhasil dikembangkan dengan baik pada perangkat lunak bernama Leap Motion Controller. Alat ini berfungsi sebagai alternatif kontrol input dalam interaksi antara manusia dengan komputer tanpa menggunakan sentuhan.

Tugas akhir yang dikerjakan oleh Franky Setiawan Daldiri merupakan sebuah aplikasi terapi stroke yang dipadukan dengan permainan simulasi seorang astronot pada luar angkasa [11]. Aplikasi itu dibangun menggunakan Unity dengan Leap Motion SDK. Dengan harapan pengguna dapat melakukan terapi secara berkesinambungan dengan tidak bosan karena terapi pascastroke merupakan masa penting untuk memulihkan otot motorik pasien sebagai pengguna aplikasi.

Dengan konsep yang sama, aplikasi ini memiliki beberapa perbedaan dengan sistem aplikasi pada tugas akhir yang saya kerjakan dan dapat dilihat pada Tabel 2.1.

Tabel 2.1 Perbedaan dengan aplikasi sejenis

No	Perbedaan	Terapi sederhana yang sudah ada	Terapi yang digunakan pada tugas akhir ini
1	Terapi dipadukan dengan games	✓	✗
2	Skenario terapi dilakukan secara berurutan tanpa terputus	✓	✗
3	Grafik ditampilkan secara langsung di aplikasi	✗	✓
4	Tiap pasien memiliki akun	✗	✓

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Bab ini menjelaskan tentang analisis dan perancangan aplikasi terapi gerak fungsional tangan pasca stroke dengan memanfaatkan *Leap Motion Controller*. Pembahasan yang akan dilakukan meliputi analisis kegunaan yang dibutuhkan dan perancangan perangkat lunak.

3.1 Analisis Perangkat Lunak

Sub bab ini menjelaskan tentang hasil analisis kebutuhan perangkat lunak serta arsitektur aplikasi terapi gerak fungsional tangan pasca stroke dengan memanfaatkan *Leap Motion Controller*. Tiap-tiap sub bab menjelaskan tentang deskripsi umum perangkat lunak, spesifikasi kebutuhan perangkat lunak, analisis aktor, arsitektur perangkat lunak, dan skenario kasus penggunaan.

3.1.1 Deskripsi Umum Perangkat Lunak

Pada Tugas Akhir ini dilakukan pengembangan aplikasi terapi gerak fungsional tangan pasca stroke dengan memanfaatkan teknologi *Leap Motion Controller*. Aplikasi ini bertujuan bagi pengguna pasca stroke yang tidak mengalami kelumpuhan tangan total tetapi pengguna yang masih memiliki keterbatasan gerak serta memiliki kekuatan dalam menggerakkan tangan yang masih perlu dilatih secara periodik untuk dapat kembali normal. Dengan *Leap Motion Controller* maka pengguna dapat menggerakkan tangan secara virtual oleh deteksi tangan dari *Leap Motion Controller*. Terdapat beberapa hal yang diperlukan agar pengguna dapat menggunakan aplikasi ini. Pertama pengguna harus memastikan bahwa komputer yang dipakai sesuai dengan spesifikasi yang dibutuhkan. Perangkat lain yang harus disediakan yaitu *Leap Motion Controller* dan seorang admin sebagai pembimbing dengan keadaan sehat. Dengan mekanisme yang telah diuraikan, aplikasi

ini dapat difungsikan sebagai aplikasi untuk terapi gerak fungsional tangan pengguna dengan memanfaatkan *Leap Motion Controller*.

3.1.2 Spesifikasi Kebutuhan Fungsional Perangkat

Kebutuhan fungsional berisi proses-proses yang harus dimiliki sistem. Kebutuhan fungsional mendefinisikan layanan yang harus disediakan dan reaksi sistem terhadap masukan pengguna. Daftar kebutuhan fungsional dapat dilihat pada Tabel 3.1.

Tabel 3.1 Daftar Spesifikasi Kebutuhan Fungsional Perangkat Lunak

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
F-0001	Melakukan pelatihan	Pengguna dapat melakukan latihan terapi stroke dimulai dari tahap pertama sampai terakhir sesuai instruksi sistem.
F-0002	Melihat hasil latihan	Pengguna dapat melihat hasil latihan berupa grafik 10 record latihan terakhir.

3.2 Perancangan Perangkat Lunak

Sub bab ini membahas bagaimana rancangan dari aplikasi Tugas Akhir ini. Perancangan perangkat lunak akan membahas model kasus penggunaan, definisi aktor, definisi kasus

penggunaan, arsitektur umum sistem, rancangan antarmuka aplikasi, dan rancangan proses aplikasi.

3.2.1 Model Kasus Penggunaan

Berdasarkan analisis spesifikasi kebutuhan fungsional dan analisis aktor dari sistem dibuat kasus penggunaan sistem. Kasus-kasus penggunaan dalam sistem ini akan dijelaskan secara rinci pada sub bab ini. Kasus penggunaan digambarkan dalam sebuah diagram kasus penggunaan. Diagram kasus penggunaan dapat dilihat pada Tabel 3.2.

Tabel 3.2 Daftar Kasus Penggunaan

Kode Kasus Penggunaan	Nama	Aktor
UC-0001	Melakukan pelatihan	Pengguna
UC-0002	Melihat hasil latihan	Admin dan Pengguna

3.2.2 Definisi Aktor

Aktor yang terlibat dalam aplikasi terapi gerak dengan memanfaatkan *Leap Motion Controller* ada dua yaitu pengguna sebagai pasien dan admin sebagai pendamping yang berada dalam kesehatan yang normal. Deskripsi pengguna secara lengkap dapat dilihat pada Tabel 3.3.

Tabel 3.3 Deskripsi Pengguna

Nama	Deskripsi
Pengguna	Merupakan aktor sebagai pasien yang bertugas untuk menjalankan aplikasi sebagai cara untuk menjalankan terapi
Admin	Merupakan aktor yang membantu pengguna dalam

	menjalankan aplikasi, tugas admin adalah memberikan inputan <i>mouse</i> pada aplikasi untuk memulai tahapan terapi dan memberikann saran sesuai nilai yang dikeluarkan aplikasi pada setiap akhir tahapan
--	--

3.2.3 Rancangan Skenario Terapi

Rancangan skenario terapi akan mengacu pada latihan gerak fungsional yang dicantumkan dalam dasar teori terapi fungsiona. Adapun latihan gerak fungsional tersebut antara lain :

1. Memegang benda
2. Mencapai ketinggian tertentu
3. Makan dengan tangan
4. Menyisir rambut
5. Mencopot baju atau jaket

Namun, dari saran latihan gerak fungsional yang sudah dipaparkan, tidak semua jenis latihan dapat diterapkan dalam tugas akhir ini. Sebagai contoh gerakan menyisir rambut sulit untuk diterapkan karena Leap Motion tidak bisa mendeteksi tangan saat posisi tangan berada di belakang kepala. Mencopot baju dan jaket juga tidak biao diterapkan karena cakupan Leap Motion tidak selebar saat manusia membuka baju atau jaket. Maka dari itu, terapi gerak fungsional yang dapat divirtualkan antara lain ;

1. Memegang benda
2. Mencapai ketinggian
3. Makan dengan tangan.

Gerakan yang disebutkan diatas akan menjadi dasar dari tugas akhir ini dan akan ditambahkan beberapa gerakan yang mengacu pada gerakan tersebut.

Dari 3 latihan itu akan dikembangkan menjadi 10 latihan dengan gerakan yang memiliki kesamaan dengan gerakan dari 3

latihan tersebut dan akan dibagi menjadi 2 tahap, yaitu tahap dasar dan tahap lanjut.

3.2.3.1 Tahap Dasar

Tahap dasar disini merupakan latihan dengan gerakan-gerakan yang cukup ringan dan merupakan turunan dari gerak memegang benda. Setiap skenario pada tahap ini memiliki dinding pada sekeliling area terapi. Dinding ini berfungsi untuk mencegah objek terlempar keluar dari area terapi dan objek tidak dapat diambil kembali. Gerakan tahap dasar dapat dilihat pada Tabel 3.4.

Tabel 3.4 Gerakan Tahap Dasar

No	Gerak	Penjelasan
1.	Menekan tombol Kategori gerak : memegang benda, mencapai ketinggian tertentu	Pada fungsi OnCollisionEnter() aplikasi mendeteksi apakah terjadi benturan antara jari dan tombol. Jika terjadi benturan dan posisi z pada jari kurang dari 9.3f, maka <i>timer</i> akan diaktifkan, merubah posisi tombol dan tombol akan berubah warna menjadi merah. Jika <i>timer</i> menyentuh angka 0 maka lama waktu pasien menyelesaikan skenario akan disimpan. Jika jari dilepaskan, maka <i>timer</i> akan berhenti dan mengeset ulang <i>timer</i> menjadi 3 lalu posisi tombol akan kembali seperti semula dan warna tombol kembali menjadi biru.
2.	Memegang gelas	Pada fungsi Update() aplikasi mendeteksi apakah gelas terenggam oleh tangan

	Kategori gerak : memegang benda	(GrabbableObject.grabbed_ = true). Jika iya, <i>timer</i> akan berjalan dan merubah warna gelas menjadi merah. Kemudian jika <i>timer</i> menyentuh angka 0, lama waktu pasien menyelesaikan skenario akan disimpan. Jika genggaman terlepas maka warna gelas akan kembali seperti semula dan mengeset <i>timer</i> menjadi 3 kembali.
3.	Mengangkat gelas Kategori gerak : memegang benda	Pada fungsi OnCollisionEnter() yang terdapat pada plane yang sudah ditentukan, aplikasi mendeteksi benturan antara gelas dan plane. Jika terjadi benturan, <i>timer</i> akan berjalan dan jika sudah menyentuh angka 0 maka lama pasien menyelesaikan skenario akan disimpan.
4.	Mengayunkan ikan Kategori gerak : memegang benda	Pada fungsi OnCollisionEnter() yang terdapat pada plane yang sudah ditentukan, aplikasi mendeteksi benturan antara ikan dan plane. Jika terjadi benturan, maka lama pasien menyelesaikan skenario akan disimpan .

3.2.3.2 Tahap Lanjut

Tahap lanjut disini merupakan latihan dengan gerakan-gerakan yang lebih sulit dari tahap dasar dan merupakan turunan dari gerak mencapai ketinggian tertentu dan makan dengan tangan. Setiap skenario pada tahap ini memiliki dinding pada sekeliling area terapi. Dinding ini berfungsi untuk mencegah objek terlempar keluar dari area terapi dan objek tidak dapat diambil kembali. Gerakan tahap lanjut dapat dilihat pada Tabel 3.5.

Tabel 3.5 Gerakan Tahap Lanjut

No	Gerak	Penjelasan
1.	Menyalakan lampu Kategori gerak : memegang benda	Terdapat 3 tombol yang mewakili huruf A, B dan C. Terdapat fungsi randomABC() yang akan menampilkan 3 huruf ABC secara acak (berbentuk string). Pasien diharapkan menekan tombol sesuai 3 huruf yang muncul tersebut. Setiap tombol yang tertekan akan membentuk sebuah string dari huruf tiap tombol yang ditekan yang mana nantinya akan dicocokkan dengan 3 huruf yang muncul dari fungsi randomABC(). Jika string dari fungsi randomABC() dan string dari tekanan tombol tidak sama dan jumlah string hasil tekanan lebih dari 3 karakter maka randomABC() akan mengeluarkan string baru yang harus ditekan oleh pasien. Jika pasien berhasil menekan sesuai

		string dari randomABC() maka lampu akan menyala dan lama pasien menyelesaikan skenario akan disimpan.
2.	Makan dengan sendok Kategori gerak : makan dengan tangan	Pada fungsi OnCollisionEnter(), aplikasi mendeteksi benturan antara sendok dengan <i>collider</i> yang sudah diletakan pada daerah mulut. Setiap benturan akan memicu <i>counter</i> bertambah. Jika <i>counter</i> sama dengan 3 maka lama pasien menyelesaikan skenario akan disimpan.
3.	Memindahkan buku Kategori gerak : memegang benda, mencapai ketinggian tertentu	Pada fungsi OnCollisionEnter() nama dari setiap objek yang berbenturan akan dicocokkan dengan nama dari ketiga buku. Jika nama dari objek yang berbenturan cocok maka akan memicu <i>counter</i> dari buku. Jika <i>counter</i> setiap buku ≥ 1 maka lama pasien menyelesaikan skenario akan disimpan.
4.	Minum dari gelas Kategori gerak : memegang benda, mencapai ketinggian tertentu	Pada fungsi OnCollisionEnter(), aplikasi mendeteksi benturan antara <i>collider</i> dengan gelas. Jika terjadi benturan maka akan memicu <i>timer</i> yang mana ketika menyentuh angka 0 lama pasien

		menyelesaikan skenario akan disimpan. Jika benturan terlepas maka <i>timer</i> akan diset ulang menjadi 3.
5.	Memutar kran air Kategori gerak : memegang benda	Pada fungsi Update(), aplikasi akan mendeteksi apakah kran tergenggam oleh tangan (GrabbableObject.grabbed_ = true). Jika iya, rotasi tangan yang dihasilkan oleh mainHand.PalmNormal.Roll akan ditampung dalam variabel dan diaplikasikan pada kran agar kran terlihat berputar. Jika variabel hasil rotasi < -70 maka akan terhitung 1 kali putaran dan ketikan jumlah putaran lebih dari 3 maka warna kran akan berubah menjadi kuning dan lama pasien menyelesaikan skenario akan disimpan.
6.	Membuka pintu Kategori gerak : memegang benda, mencapai ketinggian tertentu	Pada fungsi Update(), aplikasi mendeteksi apakah ganggang pintu tergenggam oleh tangan (GrabbableObject.grabbed_ = true). Jika iya, posisi z dari tangan akan disimpan dalam variabel yang mana akan digunakan untuk merotasi pintu. Jika sudut antara posisi pintu normal dan posisi pintu terbuka > 50° maka lama pasien

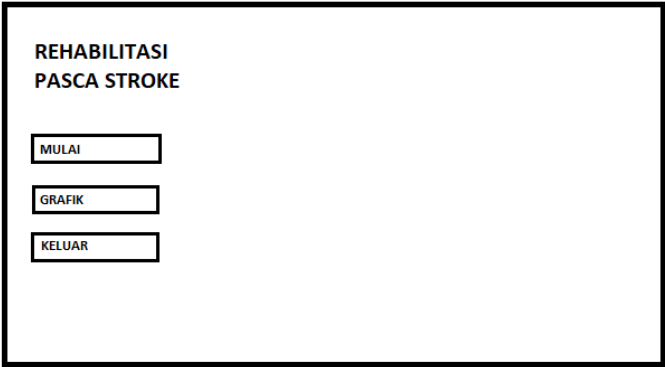
		menyelesaikan skenario akan disimpan.
--	--	---------------------------------------

3.2.4 Rancangan Antarmuka

Rancangan antarmuka aplikasi diperlukan untuk memberikan gambaran umum kepada pengguna bagaimana sistem yang ada dalam aplikasi ini berinteraksi dengan pengguna. Pada sub bab ini akan dijelaskan rancangan aplikasi untuk menggambarkan rancangan antarmuka.

3.2.4.1 Rancangan Antarmuka Menu Utama

Halaman ini merupakan tampilan awal saat aplikasi pertama dijalankan. Pada Gambar 3.1 dan spesifikasi atribut pada menu utama dapat dilihat pada Tabel 3.6.



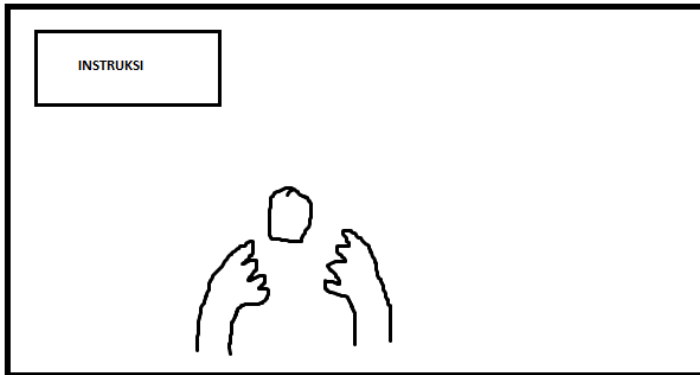
Gambar 3.1 Menu Utama

Tabel 3.6 Spesifikasi Atribut Antarmuka Menu Utama

No	Nama	Jenis	Kegunaan
1	Mulai	Button	Memulai terapi (memilih tahap)
2	Grafik	Button	Melihat grafik
3	Keluar	Button	Keluar dari aplikasi

3.2.4.2 Rancang Antarmuka Tahap Terapi

Halaman ini merupakan tampilan default pengguna memulai tahap terapi. Pada Gambar 3.2 dan spesifikasi atribut pada menu utama dapat dilihat pada Tabel 3.7.



Gambar 3.2 Tahap Terapi

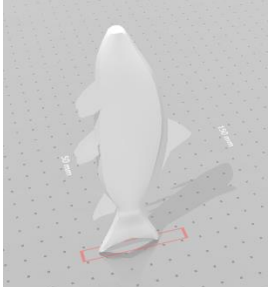
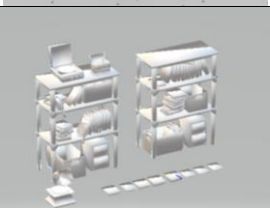

Tabel 3.7 Spesifikasi Atribut Antarmuka Tahap Terapi


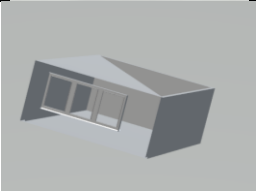


No	Nama	Jenis	Kegunaan
1	Instruksi	Text	Berisi instruksi dari tahap terapi
2	Tangan dan sekitarnya	<i>Leap Motion Controller</i> dan objek-objek pendukung terapi	Pendukung terapi

3.2.4.3 Model 3D

Model 3D diambil dari *standard asset* Unity dan website penyediaan model 3D gratis yang dapat dilihat pada Tabel 3.8.

Tabel 3.8 Daftar Objek

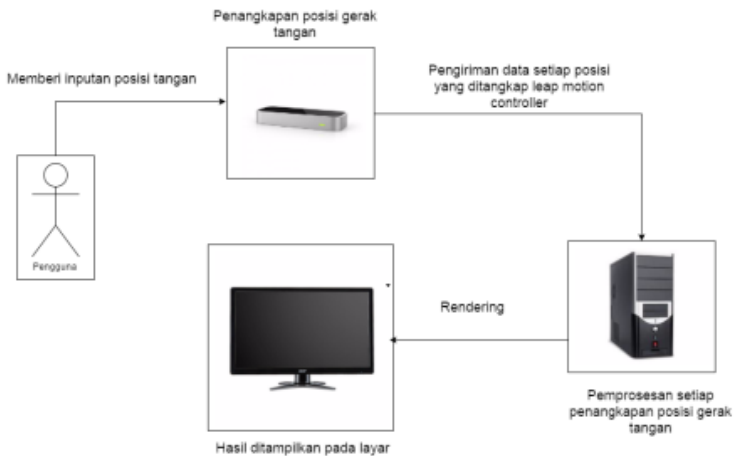
No	Gambar	Deskripsi	Sumber
1		Kran Air	https://free3d.com/
2		Ikan	https://free3d.com/
3		Rak Buku dan Buku	Unity Asset Store
4		Perabot Rumah	Unity Asset Store

5		Pintu	Unity Asset Store
6		Perabot Rumah	Unity Asset Store
7		Kucing	Unity Asset Store
8		Lampu	Unity Asset Store

9		Pipa	Unity Asset Store
10		Mangkuk	https://free3d.com/
11		Sendok	https://free3d.com/

3.2.5 Arsitektur Umum Sistem

Arsitektur sistem pada aplikasi terapi gerak tangan dengan memanfaatkan *Leap Motion Controller* ini didukung oleh beberapa perangkat yaitu komputer dan *Leap Motion Controller*. Implementasi aplikasi juga memanfaatkan salah satu aplikasi *game engine* yang sudah terkenal keunggulannya yaitu Unity. Arsitektur secara umum aplikasi ini terlihat pada Gambar 3.3.



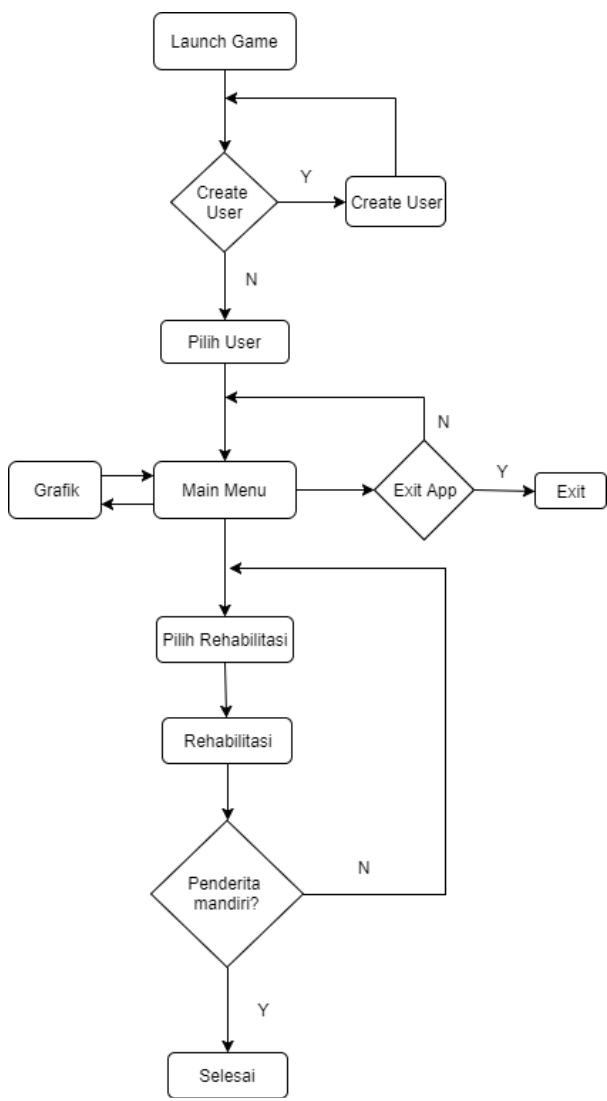
Gambar 3.3 Rancangan Sederhana Arsitektur Aplikasi

3.2.6 Rancang Proses Sistem

Proses sistem dalam aplikasi ini terbagi menjadi 2 bagian yaitu proses sistem dalam Unity dan proses sistem dalam Laravel yang akan dijelaskan dalam subbab berikut

3.2.6.1 Rancang Proses Sistem Dalam Unity

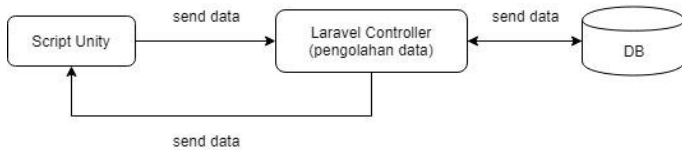
Rancang proses sistem dalam Unity dapat ditunjukkan dengan *flowchart* pada Gambar 3.4.



Gambar 3.4 Flowchart Aplikasi

3.2.6.2 Rancang Proses Sistem Dalam Laravel

Rancang proses sistem dalam Unity dapat ditunjukkan dengan *flowchart* pada Gambar 3.5.



Gambar 3.5 Flowchart Laravel

(Halaman ini sengaja dikosongkan)

BAB IV

IMPLEMENTASI

Bab ini akan menjelaskan tentang implementasi Tugas Akhir berdasarkan rancangan perangkat lunak yang diuji oleh penulis. Proses implementasi mengacu pada rancangan perangkat yang telah dilakukan sebelumnya, namun juga dimungkinkan terjadinya perubahan-perubahan jika dirasa perlu. Implementasi dilakukan dengan menggunakan bahasa C# dan php.

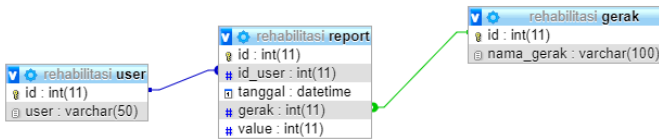
4.1 Lingkungan Implementasi Perangkat Lunak

Sub bab ini menjelaskan tentang lingkungan implementasi perangkat lunak yang dibangun. Lingkungan selama proses implementasi aplikasi terapi gerak pasca stroke dengan menggunakan *Leap Motion Controller* sebagai berikut :

Perangkat keras	<ul style="list-style-type: none">❖ Prosesor Intel(R) Core(TM) i3-2120 CPU @ 3.30GHz (4 CPUs), ~3.3GHz❖ Memory 8GB❖ Monitor❖ Leap Motion Controller
Perangkat Lunak	<ol style="list-style-type: none">1. Sistem Operasi Windows 10 Pro 64-bit2. Unity 5.5.6f13. Laravel 5.54. Leap Motion SDK5. Adobe Illustrator

4.2 Implementasi Skenario Terapi dan Antarmuka

Sub bab ini akan menjelaskan tentang implementasi dari skenario terapi dan antarmuka yang digunakan. Aplikasi ini terbagi menjadi 2 yaitu non-terapi dan terapi. Model data yang digunakan dapat dilihat pada Gambar 4.1.



Gambar 4.1 Model Data

4.2.1 Implementasi Non-terapi

Membahas implementasi yang tidak berkaitan dengan tahap terapi.

4.2.1.1 Integrasi Leap Motion dengan Unity

Leap Motion dan Unity membutuhkan penghubung dan penghubung itu adalah SDK Leap Motion. SDK Leap Motion diunduh melalui website resmi Leap Motion yaitu <https://developer.leapmotion.com/>. Hasil unduh berupa package yang nantinya akan di impor ke Unity. Hasil import berupa seperti pada Gambar 4.2.



Gambar 4.2 Hasil import Package

4.2.1.2 Pilih User

Pada tampilan ini, admin memilih akun pengguna untuk memulai latihan dan jika pengguna belum terdaftar, admin bisa membuat akun pengguna. Tampilan antarmuka pilih user dapat dilihat pada Gambar 4.3 dan tampilan membuat akun pengguna dapat dilihat pada Gambar 4.4



Gambar 4.3 Antarmuka Pilih User

Untuk mendapatkan data user, dipanggilah *function* `get_user` seperti pada Kode Sumber 4.1.

```
1. $data = DB::table('user')->get();
2. return view('index', compact('data'));
```

Kode Sumber 4.1 *Function* get_user

Lalu dilakukan *generate button* dalam antarmuka pilih user yang mana tiap buttonnya terdapat id tiap-tiap pengguna. Jika button diklik, maka kode akan menyimpan id pengguna untuk digunakan dalam proses-proses selanjutnya. Pada Kode Sumber 4.2, kode mengakses *function* `get_user` dari Laravel untuk mendapatkan semua data user.

```

1. WWW data =
2. new WWW ("http://localhost/rehabilitasi/public/rehabilitasi/user");
3.
4. yield return data;
5. string userdata = data.text;
6.
7. user = userdata.Split (';');

```

Kode Sumber 4.2 Mengambil Data

Data dari *function* `get_user` belum bisa digunakan langsung oleh Unity maka diperlukan proses olah data untuk menciptakan array dari data tersebut. Kode dapat dilihat pada Kode Sumber 4.3.

```

1. string GetData (string data, string index) {
2.     string value =
3.     data.Substring (
4.     data.IndexOf(index)+index.Length);
5.
6.     if (value.Contains ("|"))
7.         value = value.Remove (
8.         value.IndexOf("|"));
9.
10.    return value;
11. }

```

Kode Sumber 4.3 Mengolah Data

Dari array tersebut dilakukan perulangan untuk *generate* button yang mana tiap button memiliki id tiap user. Kode dapat dilihat pada Kode Sumber 4.4.

```

1.  for (int i = 0; i < max; i++){
2.      string nama =
3.          (GetData (user [i], "User:"));
4.
5.      string id = (GetData (user [i], "ID:"));
6.      string[] level = new string[12];
7.
8.      for (int x = 1; x < 12; x++){
9.
10.         level[x] =
11.             (GetData (
12.                 user [i], "Gerak "+x+":"));
13.     }
14.
15.     GameObject button =
16.         (GameObject)Instantiate (prefabButton);
17.
18.     button.transform.SetParent (
19.         parentButton.transform);
20.
21.     button.transform.GetChild (0)
22.         .GetComponent<Text> ().text = nama;
23.
24.     button.GetComponent<Button> ().onClick.
25.         AddListener (() => ButtonClicked
26.             (id, nama, level));
27. }

```

Kode Sumber 4.4 Generate Button

Tiap button memiliki *function onclick* yang mana akan menyimpan id user yang dipilih dan pindah menuju antarmuka menu utama. Kode dapat dilihat pada Kode Sumber 4.5.

```

1. public void ButtonClicked
2. (string id, string nama, string[] level){
3.     idActive = id;
4.     userActive = nama;
5.     gerak = level;
6.     SceneManager.LoadScene ("Main Menu");
7. }

```

Kode Sumber 4.5 Menyimpan Id User



Gambar 4.4 Antarmuka Menambah User Baru

Jika menambahkan user baru kode sumber dapat dilihat pada Kode Sumber 4.6.

```

- $data = DB::table('user')-
  >insert(['user' => $request->user]);
-
- if($data) return "berhasil";
- else "gagal";

```

Kode Sumber 4.6 *function create*

4.2.1.3 Menu Utama

Pada tampilan ini terdapat tiga tombol yaitu “mulai” yang berfungsi untuk memulai latihan. Tombol “grafik” yang berfungsi untuk melihat grafik latihan dan tombol “keluar” yang digunakan untuk keluar dari program. Tampilan antarmuka menu utama dapat dilihat pada Gambar 4.5



Gambar 4.5 Antarmuka Menu Utama

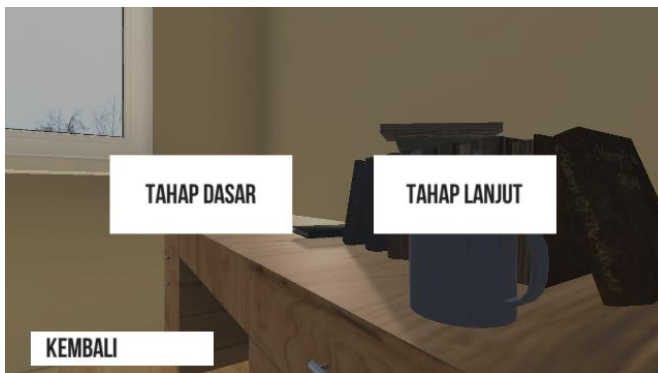
Tampilan ini hanya berisi kode untuk berpindah *scene* pada Unity. Kode dapat dilihat pada Kode Sumber 4.7.

```
1. SceneManager.LoadScene ("nama scene");
```

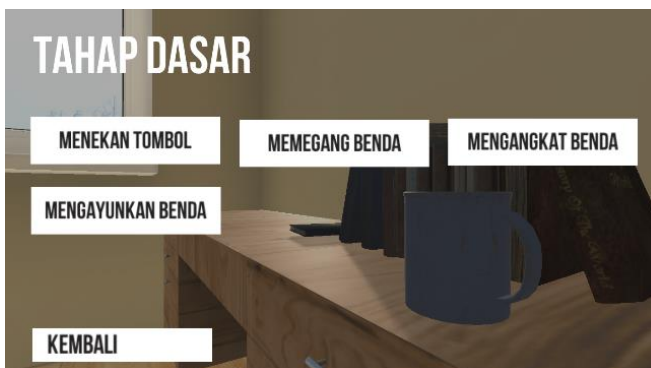
Kode Sumber 4.7 Load Scene

4.2.1.4 Pilih Level

Pada tampilan ini terdapat 3 tombol yaitu tombol “tahap dasar” untuk menuju tampilan pilihan tahap-tahap dasar, tombol “tahap lanjut” untuk menuju tampilan pilihan tahap-tahap lanjut dan tombol “kembali” untuk kembali ke menu utama. Tampilan pilih level dapat dilihat pada Gambar 4.6. Tampilan tahap dasar dapat dilihat pada Gambar 4.7. Tampilan tahap lanjut dapat dilihat pada Gambar 4.8



Gambar 4.6 Antarmuka Pilih Level



Gambar 4.7 Antarmuka Pilih Level Tahap Dasar

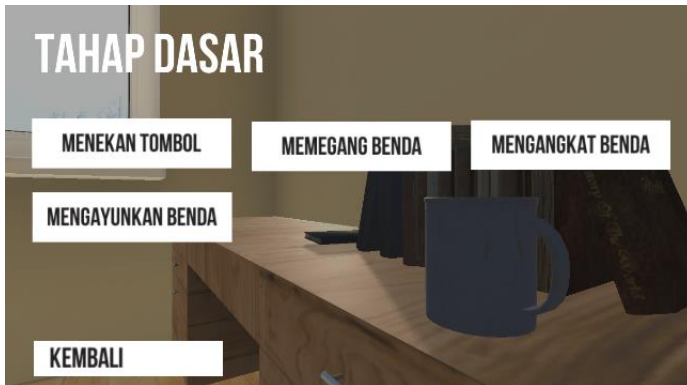


Gambar 4.8 Antarmuka Pilih Level Tahap Lanjut

Tampilan ini hanya berisi kode untuk berpindah *scene* pada Unity. Kode dapat dilihat pada Kode Sumber 4.7.

4.2.1.5 Grafik

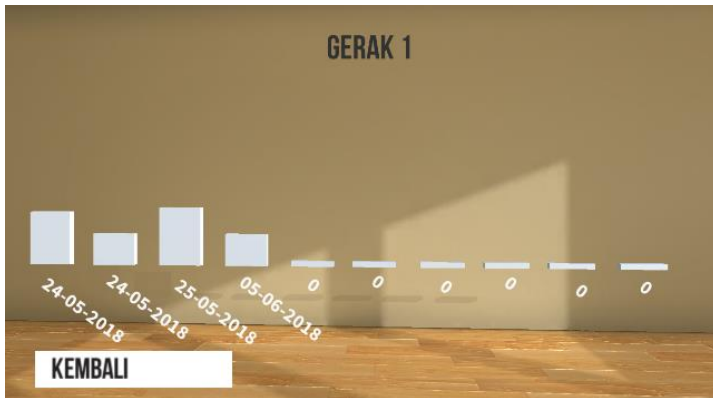
Pada tampilan ini terdapat 3 tombol yaitu tombol “tahap dasar” untuk menuju tampilan pilihan grafik tahap-tahap dasar, tombol “tahap lanjut” untuk menuju tampilan pilihan grafik tahap-tahap lanjut dan tombol “kembali” untuk kembali ke menu utama. Tampilan grafik dapat dilihat pada Gambar 4.11. Tampilan pilihan grafik tahap dasar dapat dilihat pada Gambar 4.9. Tampilan pilihan grafik tahap lanjut dapat dilihat pada Gambar 4.10.



Gambar 4.9 Antarmuka Pilihan Grafik Tahap Dasar



Gambar 4.10 Antarmuka Pilihan Grafik Tahap Lanjut



Gambar 4.11 Antarmuka Grafik

Kode akan memanggil *function* grafik2 yang digunakan untuk mengambil data grafik dari latihan yang dipilih. Kode sumber dapat dilihat pada Kode Sumber 4.8.

```

1. $data = DB::table('report')
2. ->where('id_user','=',$id)
3. ->where('gerak','=',$gerak)
4. ->limit(10)
5. ->get();
6.
7. $count = 0;
8.
9. foreach ($data as $value) {
10.     $count = $count + $value->value;
11. }
12.
13. foreach ($data as $value) {
14.     $value->value =
15.     $value->value/$count*100;
16.
17.     $value->tanggal =
18.     date_format(date_create(
19.     $value->tanggal), 'd-m-Y');
20. }
21.

```

```

22. return view('grafik2',
23. compact('data', 'gerak'));

```

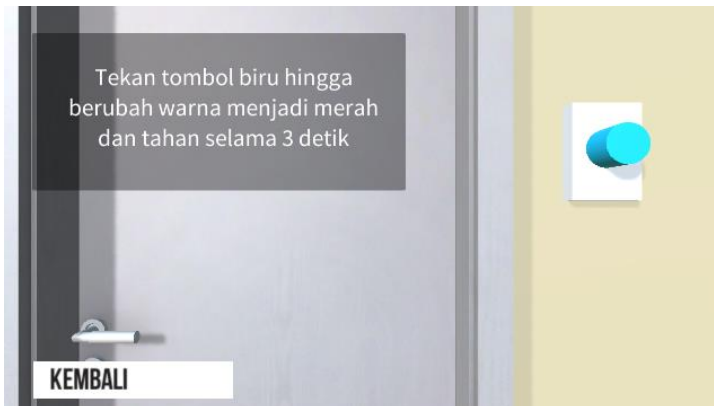
Kode Sumber 4.8 *function* grafik2

4.2.2 Implementasi Terapi

Membahas implementasi yang berkaitan dengan tahap terapi.

4.2.2.1 Menekan Tombol

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan menekan tombol. Tampilan dapat dilihat pada Gambar 4.12.



Gambar 4.12 Antarmuka Menekan Tombol

Kode mendeteksi *collision* antara jari dan tombol dapat dilihat pada Kode Sumber 4.9.

```

1. void OnCollisionEnter(Collision other){
2.     FingerModel finger = other.gameObject.
3.     GetComponentInParent<FingerModel> ();
4.
5.     if (finger) {
6.         if (finger.GetTipPosition ().x < 9.3f) {
7.             pencet = 1;
8.             StartCoroutine (ChangeColor (
9.                 gameObject,
10.                 inactiveColor,
11.                 activeColor, 0.5f));
12.         }
13.
14.         transform.localPosition = Vector3.Lerp (
15.             endPosition,
16.             startPosition,
17.             Time.deltaTime * 10);
18.     }
19. }
20.
21. void OnCollisionExit(){
22.     pencet = 0;
23.     StartCoroutine (ChangeColor (
24.         gameObject,
25.         activeColor,
26.         inactiveColor, 0.5f));
27.
28.     transform.localPosition = Vector3.Lerp (
29.         startPosition,
30.         endPosition,
31.         Time.deltaTime * 10);
32.
33.     GameObject.Find ("timer").
34.     GetComponent<Text> ().text = "Waktu : 3";
35.     targetTime = 3.0f;
36. }
37.

```

Kode Sumber 4.9 Collision jari dan tombol

Jika terjadi *collision* antara jari dan tombol *timer* akan berjalan dan jika sudah menyentuh angka 0 maka kode akan memanggil *function* untuk menyimpan. Kode dapat dilihat pada Kode Sumber 4.10.

```

1. void Update()
2. {
3.     if (pencet == 1) {
4.         targetTime -= Time.deltaTime;
5.         if (targetTime > 0) {
6.             GameObject.Find ("timer").
7.             GetComponent<Text> ().text =
8.             "Waktu : " + Mathf.RoundToInt
9.             (targetTime).ToString ();
10.        } else {
11.            GameObject.Find ("timer").
12.            GetComponent<Text> ().text =
13.            "Waktu : 0";
14.        }
15.
16.        if (targetTime < 1) {
17.            simpan = 1;
18.            if (simpan == 1) {
19.                simpan++;
20.                Instruksi.SetActive (false);
21.                Panel.SetActive (true);
22.                GameObject.Find ("timer").
23.                GetComponent<Text> ().text =
24.                "Waktu : 0";
25.            }
26.        }
27.    }
28. }
29.

```

Kode Sumber 4.10 Menyalakan *timer*

Function untuk menyimpan dapat dilihat pada Kode Sumber 4.11.


```

1. public void timerEnd() {
2.     WWWForm form = new WWWForm ();
3.     form.AddField
4.     ("id", GenerateButton.idActive);
5.
6.     form.AddField ("gerak", 1);
7.
8.     form.AddField
9.     ("value", realTime.ToString());
10.
11.     WWW www = new WWW ("http://localhost/rehabili
    tasi/public/rehabilitasi/update", form);
12.
13.     if (www.error != null) {
14.         print (www.error);
15.     } else {
16.         Instruksi.SetActive (false);
17.         Panel.SetActive (true);
18.     }
19. }

```

Kode Sumber 4.11 Menyimpan Hasil

Function ini digunakan untuk menginput data waktu saat selesai menyelesaikan latihan. *Function* ini digunakan saat antarmuka latihan dan kode sumber dapat dilihat pada Kode Sumber 4.12.

```

1. $data = DB::table('report')
2. ->insert([
3.     'id_user' => $request->id,
4.     'gerak' => $request->gerak,
5.     'value' => $request->value
6. ]);
7.
8. if($data) return "berhasil";
9. else "gagal";

```

Kode Sumber 4.12 *function* update

4.2.2.2 Memegang Benda

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan memegang benda. Tampilan dapat dilihat pada Gambar 4.13



Gambar 4.13 Antarmuka Memegang Benda

Kode ini digunakan pada latihan memegang benda. Kode melakukan pengecekan apakah objek sedang terpegang oleh tangan atau tidak. Kode dapat dilihat pada Kode Sumber 4.13.

```

1.  if (GrabbableObject.grabbed_ == true) {
2.      StartCoroutine (ChangeColor (
3.          gameObject,
4.          inactiveColor,
5.          activeColor,
6.          0.2f));
7.
8.      simpan += 1;
9.
10.     if (simpan == 1) {
11.         simpan += 1;
12.         timerEnd ();
13.     }

```

```
14. }
```

Kode Sumber 4.13 Mengecek Genggaman

Jika berhasil maka akan memanggil *function* untuk menyimpan dapat dilihat pada Kode Sumber 4.11 dan Kode Sumber 4.12.

4.2.2.3 Mengangkat Benda

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan mengangkat benda. Tampilan dapat dilihat pada Gambar 4.14



Gambar 4.14 Antarmuka Mengangkat Benda

Kode ini digunakan pada latihan mengangkat benda. Kode melakukan pengecekan apakah objek sudah berpindah ke posisi yang sudah ditentukan. Kode dapat dilihat pada Kode Sumber 4.14.

```
1. if (other.gameObject.name == "mainBox") {
2.     targetTime -= Time.deltaTime;
3.
4.     if (targetTime < 0) {
5.         save += 1;
```

```

6.
7.         if (save == 1) {
8.             save += 1;
9.             simpan ();
10.        }
11.    }
12. }

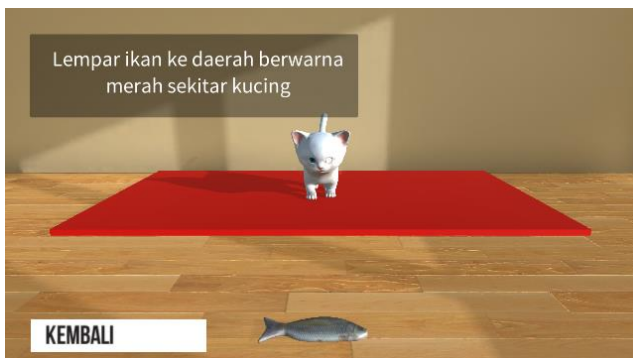
```

Kode Sumber 4.14 Mengecek Lokasi

Jika berhasil maka akan memanggil *function* untuk menyimpan dapat dilihat pada Kode Sumber 4.11 dan Kode Sumber 4.12.

4.2.2.4 Mengayunkan Benda

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan mengayunkan benda. Tampilan dapat dilihat pada Gambar 4.15.



Gambar 4.15 Antarmuka Mengayunkan Benda

Kode ini digunakan pada latihan mengayunkan benda. Kode melakukan pengecekan apakah objek sudah berpindah ke posisi yang sudah ditentukan. Kode dapat dilihat pada Kode Sumber 4.14. Jika berhasil maka akan memanggil *function* untuk

menyimpan dapat dilihat pada Kode Sumber 4.11 dan Kode Sumber 4.12.

4.2.2.5 Menyalakan Lampu

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan menyalakan lampu. Tampilan dapat dilihat pada Gambar 4.16.



Gambar 4.16 Antarmuka Menyalakan Lampu

Kode ini digunakan pada latihan menyalakan lampu. Kode akan mengenerate karakter yang mana akan diinput ulang oleh user. Karakter hanya sebatas huruf 'A', 'B' dan 'C'. Kode dapat dilihat pada Kode Sumber 4.15.

```

1. string abc = "ABC";
2.
3. string random =
4. abc [Random.Range (0,abc.Length)] +
5. ""+
6. abc [Random.Range (0,abc.Length)] +
7. ""+
8. abc [Random.Range (0,abc.Length)];
9.
10. return random;

```

Kode Sumber 4.15 Men-generate karakter

Setelah men-*generate* karakter, akan terbentuk sebuah string dan akan ditampilkan di layar. Kode akan mendeteksi posisi x, y dan z dari tangan. Kode dapat dilihat pada Kode Sumber 4.9.

User melakukan inputan sesuai string yang muncul. Kode mengecek posisi x, y dan z untuk mengecek inputan seperti pada Kode Sumber 4.9. Setiap inputan akan membentuk string seperti pada Kode Sumber 4.16.

```

1. if (other.gameObject.name == "A") {
2.     Gerak5_3.text = Gerak5_3.text + "A";
3. } else if (other.gameObject.name == "B") {
4.     Gerak5_3.text = Gerak5_3.text + "B";
5. } else if (other.gameObject.name == "C") {
6.     Gerak5_3.text = Gerak5_3.text + "C";
7. }
```

Kode Sumber 4.16 Input Karakter

String inputan user dan string hasil *generate* akan dicocokkan. Jika sama user telah berhasil menyelesaikan latihan dan akan memanggil *function* untuk menyimpan hasil seperti pada Kode Sumber 4.11 dan Kode Sumber 4.12. Kode dapat dilihat pada Kode Sumber 4.17.

```

1. if (text == jawaban) {
2.     Lamp_OFF.SetActive (false);
3.     Lamp_ON.SetActive (true);
4.
5.     text = "";
6.     save += 1;
7.     if (save == 1) {
8.         save += 1;
9.         simpan ();
10.    }
11.
12.    Instruksi.SetActive (false);
13.    Panel.SetActive (true);
```

```

14. } else if (text != jawaban &&
15. text.Length == jawaban.Length ||
16. text.Length > jawaban.Length) {
17.     SceneManager.LoadScene ("Gerak 5");
18. }

```

Kode Sumber 4.17 Pengecekan String

4.2.2.6 Makan Dengan Sendok

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan makan dengan sendok. Tampilan dapat dilihat pada Gambar 4.17



Gambar 4.17 Antarmuka Makan Dengan Sendok

Kode ini digunakan pada latihan makan dengan sendok. Setiap sendok menyentuh bagian mulut maka counter akan bertambah dan jika counter sudah mencapai 3 maka akan memanggil *function* untuk menyimpan hasil seperti pada Kode Sumber 4.11 dan Kode Sumber 4.12. Kode dapat dilihat pada Kode Sumber 4.18.

```

1. void Update () {
2.     realTime += Time.deltaTime;

```

```

3.     if (count == 3) {
4.         save += 1;
5.         if (save == 1) {
6.             save += 1;
7.             simpan ();
8.         }
9.     }
10. }
11.
12. void OnCollisionEnter(Collision other){
13.     if (other.gameObject.name == "Plane002"){
14.         sereal = GameObject.Find (
15.             "serealdisendok");
16.
17.         sereal.SetActive (false);
18.         count++;
19.     }
20. }

```

Kode Sumber 4.18 Cek *Collision* Sendok

4.2.2.7 Memindah Benda

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan memindah benda. Tampilan dapat dilihat pada Gambar 4.18.



Gambar 4.18 Antarmuka Memindah Benda

Kode ini digunakan pada latihan memindahkan benda. User harus memindahkan 3 buah buku pada lokasi yang sudah ditentukan. Kode dapat dilihat pada Kode Sumber 4.19. Jika berhasil kode akan memanggil *function* untuk menyimpan data seperti pada Kode Sumber 4.11 dan Kode Sumber 4.12.

```

1. void Update () {
2.     realTime += Time.deltaTime;
3.
4.     if (count1 == 1 && count2 == 1 &&
5.         count3 == 1) {
6.         targetTime -= Time.deltaTime;
7.         if (targetTime <= 0.0f) {
8.             save += 1;
9.             if (save == 1) {
10.                 save += 1;
11.                 simpan ();
12.             }
13.         }
14.     }
15. }
16.
17. void OnCollisionEnter(Collision other){
18.     if (other.gameObject.name == "book1")
19.         count1++;
20.     if (other.gameObject.name == "book2")
21.         count2++;
22.     if (other.gameObject.name == "book3")
23.         count3++;
24. }

```

Kode Sumber 4.19 Cek Posisi Buku

4.2.2.8 Minum Dari Gelas

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan minum dari gelas. Tampilan dapat dilihat pada Gambar 4.19.



Gambar 4.19 Antarmuka Minum Dari Gelas

Kode ini digunakan pada latihan meminum dari gelas. Jika gelas menempel pada bagian mulut maka akan memanggil *function* untuk menyimpan hasil latihan seperti pada Kode Sumber 4.11 dan Kode Sumber 4.12. kode dapat dilihat pada Kode Sumber 4.20.

```

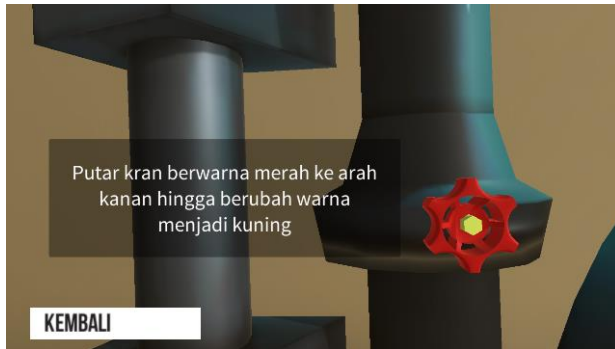
1. if (other.gameObject.name == "cup") {
2.     save += 1;
3.     if (save == 1) {
4.         save += 1;
5.         simpan ();
6.     }
7.     Panel.SetActive (true);
8. }

```

Kode Sumber 4.20 Cek Collision Gelas

4.2.2.9 Memutar Kran

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan memutar kran. Tampilan dapat dilihat pada Gambar 4.20.



Gambar 4.20 Antarmuka Memutar Kran

Kode ini digunakan dalam latihan memutar kran. Kran diputar 3 kali lalu akan memanggil *function* untuk menyimpan hasil latihan seperti pada Kode Sumber 4.11 dan Kode Sumber 4.12. Kode dapat dilihat pada Kode Sumber 4.21.

```

1. void Update()
2. {
3.     Frame frame = controller.Frame();
4.     Hand mainHand = frame.Hands.Frontmost;
5.     realTime += Time.deltaTime;
6.
7.
8.     if (GrabbableObject.grabbed_ == true) {
9.         z = mainHand.PalmNormal.Roll * 30;
10.        transform.rotation =
11.        Quaternion.Euler (0, 0, z);
12.
13.        if (z < -70) {
14.            y += 1;
15.            if (y == 1) {
16.                cek += 1;
17.                GameObject.Find ("timer").
18.                GetComponent<Text> ().text =
19.                "Putaran : " +
20.                Mathf.RoundToInt (cek).
21.                ToString ();
22.            }

```

```

23.         }
24.
25.         if (cek > 3) {
26.             StartCoroutine (ChangeColor (
27.                 gameObject,
28.                 inactiveColor,
29.                 activeColor, 0.3f));
30.             save += 1;
31.             if (save == 1) {
32.                 save += 1;
33.                 timerEnd ();
34.             }
35.         }
36.     } else {
37.         y = 0;
38.         x = 0;
39.     }
40. }
41.

```

Kode Sumber 4.21 Putar Kran

4.2.2.10 Membuka Pintu

Pada tampilan ini terdapat panel instruksi, tombol kembali dan objek untuk latihan membuka pintu. Tampilan dapat dilihat pada Gambar 4.21.



Gambar 4.21 Antarmuka Membuka Pintu

Kode digunakan pada latihan membuka pintu. Kode akan mengecek posisi z tangan jika sudah menyentuh ganggang pintu. Kode akan memanggil *function* untuk menyimpan hasil jika pintu sudah terbuka lebih dari 50° seperti pada Kode Sumber 4.11 dan Kode Sumber 4.12. kode dapat dilihat pada Kode Sumber 4.22.

```
1. if(GrabbableObject.grabbed_ == true){  
2.     z = -mainHand.PalmPosition.z*0.3f;  
3.     Pintu.transform.rotation =  
4.     Quaternion.Euler (-90, 90, z);  
5.  
6.     if (z > 50) {  
7.         simpan += 1;  
8.         if (simpan == 1) {  
9.             simpan += 1;  
10.            timerEnd ();  
11.        }  
12.    }  
13. }
```

Kode Sumber 4.22 Rotasi Pintu

(Halaman ini sengaja dikosongkan)

BAB V

UJI COBA DAN EVALUASI

Bab ini membahas pengujian dan evaluasi pada aplikasi yang dikembangkan. Pengujian yang dilakukan adalah pengujian terhadap kebutuhan fungsionalitas sistem dan kegunaan sistem. Pengujian fungsionalitas mengacu pada kasus pada Bab II. Pengujian kegunaan program dilakukan dengan mengetahui tanggapan dari pengguna terhadap sistem. Hasil evaluasi menjabarkan tentang rangkuman hasil pengujian pada bagian akhir bab ini.

5.1 Lingkungan Pengujian

Lingkungan pengujian sistem pada pengerjaan Tugas Akhir ini dilakukan pada lingkungan dan alat kakas seperti yang tertera pada Tabel 5.1.

Tabel 5.1 Lingkungan Pengujian Sistem

Perangkat keras	<ul style="list-style-type: none">❖ Prosesor Intel(R) Core(TM) i3-2120 CPU @ 3.30GHz (4 CPUs), ~3.3GHz❖ Memory 8GB❖ Monitor❖ Leap Motion Controller
Perangkat Lunak	<ul style="list-style-type: none">6. Sistem Operasi Windows 10 Pro 64-bit7. Unity 5.5.6f18. Laravel 5.59. Leap Motion SDK

5.2 Pengujian Aplikasi

Pengujian aplikasi dilakukan untuk mengetahui kesesuaian keluaran dari tiap tahap atau langkah penggunaan fitur terhadap skenario yang dipersiapkan. Pengujian fungsionalitas aplikasi akan dilakukan oleh penulis. Pengujian juga dilakukan oleh seorang fisioterapi, kritik serta pendapat beliau dapat dilihat dalam sub bab 6.2 yang berisi kritik dan saran pengembangan aplikasi kedepan.

5.2.1 Skenario Pengujian Fungsionalitas

Pada sub bab ini dijelaskan beberapa skenario uji coba perangkat lunak secara mandiri berdasarkan metode kotak hitam sebagai dasar tolok ukur keberhasilan. Pengujian fungsionalitas yang terdapat pada aplikasi dijabarkan sebagai berikut :

1. Uji coba membuat akun
2. Uji coba latihan terapi
3. Uji coba melihat grafik

Daftar uji coba tersebut merupakan pengujian yang dilakukan untuk menguji setiap kasus penggunaan pada perangkat lunak yang dibangun. Dibuat beberapa skenario yang dilakukan pada setiap daftar pengujian tersebut. Penjelasan mengenai cara dan hasil pengujian fungsionalitas perangkat lunak dibahas pada sub bab hasil uji coba.

5.2.2 Hasil Uji Coba Fungsionalitas

Pada sub bab ini dijelaskan secara detail mengenai skenario yang dilakukan dan hasil yang didapatkan dari pengujian fungsionalitas perangkat lunak yang dibangun. Penjelasan disajikan dengan menampilkan kondisi awal, masukan, keluaran, hasil yang dicapai dan kondisi akhir. Berikut ini merupakan penjabaran skenario dan hasil pengujian yang dicapai pada tiap-tiap fungsionalitas perangkat lunak.

5.2.2.1 Uji Coba Membuat Akun

Uji coba latihan ini berfungsi untuk mengetahui keberhasilan aplikasi dalam membuat akun baru bagi pasien penderita pasca stroke yang dilakukan oleh pendamping pasien. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.2.

Tabel 5.2 Uji Coba Membuat Akun

ID	UJ-P-01
Nama	Uji coba membuat akun
Tujuan Uji Coba	Pendamping pasien dapat membuat akun untuk pasien
Kondisi Awal	Aplikasi sudah terbuka
Skenario 1	Pengguna membuat akun
Keluaran yang diharapkan	Nama akun yang baru diinputkan muncul dalam list nama-nama akun
Hasil uji coba	Berhasil
Kekurangan	Akan lebih baik jika setelah mendaftar akun langsung terpilih
Kondisi akhir	Nama akun yang baru diinputkan muncul dalam list nama-nama akun

5.2.2.2 Uji Coba Latihan Terapi

Uji coba latihan ini berfungsi untuk mengetahui keberhasilan aplikasi dalam menjalankan terapi anggota gerak tangan untuk pengguna sebagai pasien pasca stroke. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.3.

Tabel 5.3 Uji Coba Latihan Terapi

ID	UJ-P-02
Nama	Uji coba latihan terapi
Tujuan Uji Coba	Skenario terapi yang sudah dibuat memungkinkan untuk dijalankan oleh pasien
Kondisi Awal	Pendamping sudah masuk pada akun pasien dan Leap Motion sudah terpasang
Skenario 1	Pendamping memilihkan gerak yang ingin dijalankan oleh pasien
Keluaran yang diharapkan	Setiap selesai menjalankan latihan akan muncul ucapan selamat
Hasil uji coba	Berhasil
Kekurangan	Gerak tangan kadang terlalu sensitif atau tidak sesuai dengan gerak asli sehingga tidak memungkinkan menyelesaikan gerakan tertentu
Kondisi akhir	Setiap selesai menjalankan latihan akan muncul ucapan selamat

5.2.2.3 Uji Coba Melihat Grafik

Uji coba latihan ini berfungsi untuk mengetahui keberhasilan aplikasi dalam menampilkan grafik. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.4.

Tabel 5.4 Uji Coba Melihat Grafik

ID	UJ-P-03
Nama	Uji coba membuat akun

Tujuan Uji Coba	Pendamping dapat melihat grafik pasien
Kondisi Awal	Pendamping sudah masuk pada akun pasien
Skenario 1	Pengguna membuat akun dan pasien sudah pernah melakukan beberapa latihan gerak (tidak harus semua)
Keluaran yang diharapkan	Aplikasi akan menampilkan grafik dari latihan gerak yang dipilih
Hasil uji coba	Berhasil
Kekurangan	Diberi sedikit penjelasan mengenai bentuk grafik, mana yang menunjukkan progress baik apakah grafik naik atau grafik turun
Kondisi akhir	Aplikasi akan menampilkan grafik dari latihan gerak yang dipilih

5.3 Konsultasi Dengan Ahli Fisioterapi

Konsultasi dengan ahli bersama Niniek Soetini, M.Fis di klinik fisioterapi RS Siloam pada alamat Jl. Sumatra 98b Kota Surabaya. Konsultasi dilakukan dengan pengujian dan wawancara dengan ahli mengenai aplikasi yang sudah diselesaikan. Kesimpulan yang diambil dari wawancara tersebut adalah aplikasi ini sudah cukup baik namun tidak bisa dijadikan terapi utama bagi pasien pasca stroke karena hanya bersifat virtual sedangkan pasien butuh merasakan masa. Aplikasi ini hanya bisa dijadikan selingan atau permainan bagi pasien.

5.4 Evaluasi Pengujian

Berdasarkan pengujian aplikasi dan pengujian pengguna yang telah dijelaskan pada subbab sebelumnya, maka didapat evaluasi sebagai berikut :

1. Aplikasi berhasil dalam tahap membuat akun dimana hal ini dibuktikan pada UJ-P-01. Namun, ada sedikit masukan untuk tahap ini yaitu aplikasi langsung menuju menu utama setelah membuat akun tanpa perlu memilihnya.
2. Aplikasi berhasil melakukan latihan terapi gerak pasca stroke bagi pasien namun tidak sepenuhnya berhasil. Keberhasilan tergantung pada kondisi Leap Motion. Jika Leap Motion dalam kondisi yang baik, gerak tangan yang muncul dalam layar sedikit sesuai dengan kondisi asli. Jika Leap Motion dalam kondisi tidak baik, gerak tangan sangat tidak akurat. Dijelaskan pada Tabel 5.5.
3. Aplikasi berhasil menampilkan grafik terbukti pada pengujian UJ-P-03.
4. Keberhasilan fungsional aplikasi tidak menjadi tolak ukur keberhasilan aplikasi dalam membantu pasien pulih.

Tabel 5.5 Hasil Evaluasi Skenario

No.	Skenario	Evaluasi
1	Menekan tombol, Menyalakan lampu	<ul style="list-style-type: none"> • Beberapa bentuk tangan tidak terdeteksi jenis jarinya, seperti bentuk tangan menunjuk • Sulit untuk stabil dalam menekan karena pergerakan tangan cukup sensitif sehingga kondisi menekan tombol terkadang sulit tercapai

2	<p>Scene yang mengandung attribute <i>IsKinematic</i> seperti:</p> <ul style="list-style-type: none"> • Memegang gelas • Mengangkat gelas • Mengayunkan ikan • Minum dari gelas • Makan dengan sendok • Memindahkan buku 	<ul style="list-style-type: none"> • Jika gelas dalam kondisi tergegangam, benda bisa dipaksa menembus dinding pembatas sehingga jika benda terjatuh tidak bisa dikembalikan lagi ke area terapi • LeapMotion_CoreAssets-3.0.0 memiliki kondisi dimana ketika tangan menggenggam, <i>rigid body</i> terdekat akan tertarik menuju tangan sehingga jika sensitifitas sedang tinggi ada peluang gelas terlempar
3	Menyalakan lampu	<ul style="list-style-type: none"> • Salah satu bentuk jari tangan yang terdeteksi jenis jarinya (tangan terbuka) memungkinkan terjadinya 2 tombol aktif secara bersamaan karena lebar tangan menyentuh 2 tombol
4	Memindahkan buku	<ul style="list-style-type: none"> • Posisi kamera mempengaruhi <i>hand gesture</i> dari gerak tangan, jika kamera terlalu tinggi, <i>hand gesture</i> tidak sesuai dengan gerak tangan asli
5	Memutar kran air	<ul style="list-style-type: none"> • Tidak ada kendala
6	Membuka pintu	<ul style="list-style-type: none"> • Tidak ada kendala

(Halaman ini sengaja dikosongkan)

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini akan diberikan kesimpulan yang diambil selama pengerjaan Tugas Akhir serta saran-saran tentang pengembangan yang dapat dilakukan terhadap Tugas Akhir ini di masa yang akan datang.

6.1 Kesimpulan

Dari hasil selama proses perancangan, implementasi, serta pengujian dapat diambil kesimpulan bahwa aplikasi tidak bisa dijadikan terapi utama bagi pasien pasca stroke karena hanya bersifat virtual sedangkan pasien membutuhkan masa untuk melatih otot.

6.2 Saran

Berikut saran-saran untuk pengembangan dan perbaikan sistem di masa yang akan datang. Diantaranya adalah sebagai berikut:

1. Memperbaiki tampilan agar lebih menarik bagi pengguna.
2. Melakukan penelitian agar aplikasi mempunyai kemampuan dimana pasien dapat merasakan beban benda secara nyata, tidak hanya virtual.

(Halaman ini sengaja dikosongkan)

DAFTAR PUSTAKA


- [1] I. Glory Acupuncture, “gloryacupuncture,” L.Ac. Joanna, 27 Maret 2012. [Online]. Available: <http://www.gloryacupuncture.com>. [Diakses 16 April 2018].
- [2] A. Krishnan, “quora,” 30 October 2015. [Online]. Available: <https://www.quora.com/Is-there-any-cure-for-a-brain-stroke>. [Diakses 03 January 2018].
- [3] M. David F. Clark, “sharecare,” [Online]. Available: <https://www.sharecare.com/health/stroke-treatment/is-there-cure-stroke>. [Diakses 03 January 2018].
- [4] M. Niniek Soetini, Interviewee, *Rehabilitasi Pasca Stroke*. [Wawancara]. 23 June 2018.
- [5] D. Springer, “Springer Link,” 2008. [Online]. Available: https://link.springer.com/referenceworkentry/10.1007%2F978-1-4020-5614-7_1209. [Diakses 18 April 2018].
- [6] Wikipedia, “Wikipedia,” Leap Motion, Inc., 01 November 2010. [Online]. Available: https://en.wikipedia.org/wiki/Leap_Motion. [Diakses 30 December 2017].
- [7] Wikipedia, “Wikipedia,” Unity, 08 June 2005. [Online]. Available: [https://en.wikipedia.org/wiki/Unity_\(game_engine\)](https://en.wikipedia.org/wiki/Unity_(game_engine)). [Diakses 30 December 2017].
- [8] S. I. Lesmana, “esaunggul,” [Online]. Available: <http://ftl221.weblog.esaunggul.ac.id/wp-content/uploads/sites/965/2015/03/Terapi-Latihan-2-Pertemuan-11.ppt>. [Diakses 03 December 2017].
- [9] T. Bhumi, “Suwekaprabha Yoga,” 27 October 2012. [Online]. Available:

<https://suwekaprabhayoga.wordpress.com/2012/10/27/gerak-anatomis/>. [Diakses 25 July 2018].

- [10] idcloudhost, "idcloudhost," idcloudhost, 07 June 2016. [Online]. Available: <https://idcloudhost.com/pengertian-dan-keunggulan-framework-laravel/>. [Diakses 04 June 2018].
- [11] F. S. Daldiri, Rancang Bangun Aplikasi Terapi Pasca Stroke untuk Latihan Pergerakan Jari Tangan dengan Menggunakan Leap Motion Controller, Surabaya: Institut Teknologi Sepuluh Nopember, 2017.

LAMPIRAN

A. Hasil Kuisisioner



Kuesioner Tugas Akhir “Rehabilitasi Pasca Stroke”

5114100114 – Shafly Naufal Adianto

IDENTITAS RESPONDEN

Nama Lengkap : NUNUK SOETIN Surabaya, 2018

Pekerjaan : PSIKOTERAPIS

Usia : 66 RL

A. KARAKTERISTIK RESPONDEN

Isilah pertanyaan di bawah ini dengan menggunakan tanda silang (X)

- 1.
2. Anakah anda mengetahui aplikasi terapi stroke ?
☒ a. Iya, Tahu ☐ b. Tidak Tahu
3. Anakah anda pernah mencoba aplikasi terapi stroke sebelumnya?
☒ a. Iya, Pernah ☐ b. Tidak Pernah

B. PENILAIAN TERHADAP APLIKASI

Isilah pertanyaan di bawah ini dengan menggunakan tanda centang (✓)

Keterangan :
 SS = Sangat setuju S = Setuju CS = Cukup setuju
 KS = Kurang setuju TS = Tidak Setuju STS = Sangat tidak setuju

No	Parameter Antar Muka	ST	T	K	CS	S	S
1.	Aplikasi memiliki tampilan, warna, dan desain yang menarik.					✓	
2.	Aplikasi ini memiliki tata letak tombol, instruksi dan informasi yang mudah dipahami					✓	
No	Parameter Fungsionalitas	ST	T	K	CS	S	S
3.	Aplikasi dapat membantu pasien pasca stroke untuk berlatih dari rumah					✓	
4.	Aplikasi dapat membantu memantau perkembangan pasien				✓		
5.	Aplikasi dapat membuat terapi menjadi lebih interaktif dan menyenangkan					✓	
No	Parameter Penilaian	ST	T	K	CS	S	S
6.	Dasar pemberian nilai pada keberhasilan gerakan pasien jelas						✓
7.	Aplikasi dapat mendeteksi gerakan pasien dan memberikan penilaian secara akurat				✓		
No	Parameter Kenyamanan	ST	T	K	CS	S	S

		S	S	S		S
8.	Aplikasi dapat berjalan lancar tanpa lag dan crash					✓
9.	Instruksi pasien untuk gerakan tidak membingungkan.				✓	
10.	Saya merasa nyaman selama menggunakan aplikasi ini			✓		

C. KRITIK DAN SARAN

Berapapun alat & perangkat yang pakai
Makanya & krap
Krap optimasi user jadi lebih baik

BIODATA PENULIS



Penulis, Shafly Naufal Adianto, lahir di kota Cilacap pada tanggal 12 April 1996. Penulis dibesarkan di Gombang, Kebumen, Jawa Tengah.

Setelah menempuh pendidikan SLTA sederajat di SMK Telkom Sandhy Putra Purwokerto tepatnya tahun 2014, penulis melanjutkan pendidikan S1 jurusan Teknik Informatika di Institut Teknologi Sepuluh Nopember Surabaya, Jawa Timur.

Di jurusan Teknik Informatika, penulis mengambil bidang minat Interaksi, Grafika dan Seni atau biasa disebut IGS dan memiliki ketertarikan di bidang website. Penulis aktif sebagai BPH 3D Schematics periode 2015 dan 2016 serta kegiatan organisasi dan event mahasiswa lainnya. Penulis dapat dihubungi melalui alamat email shaflynaufal96@gmail.com.